

AD-A174 172

A STUDY OF UPPER ERROR LIMITS IN ACCOUNTING POPULATIONS

1/2

(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH

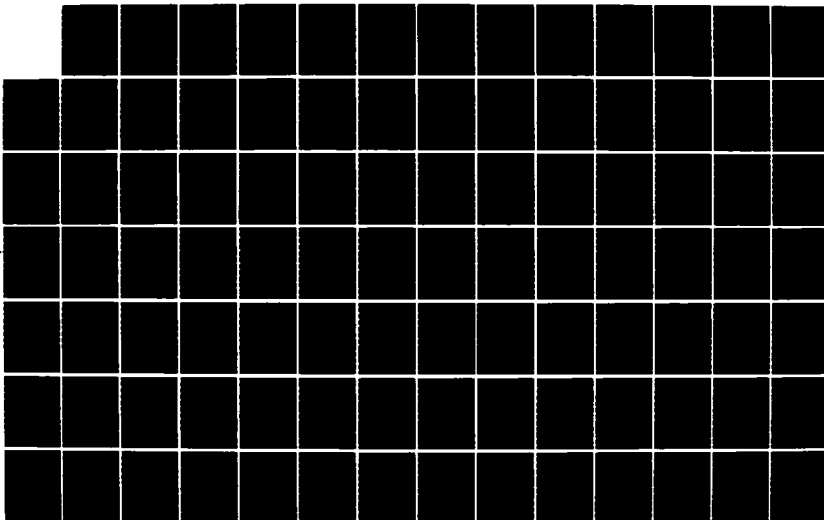
SCHOOL OF SYSTEMS AND LOGISTICS G S BRINGLE SEP 86

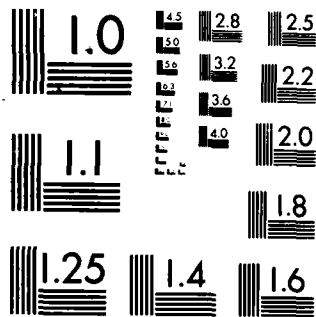
UNCLASSIFIED

AFIT/GSM/LSV/86S-4

F/G 12/1

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

2

AD-A174 172



A STUDY OF UPPER ERROR LIMITS
IN ACCOUNTING POPULATIONS

THESIS

G. Steven Bringle
Captain, USAF

AFIT/GSM/LSY/86S-4

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DTIC
ELECTE
NOV 20 1986

B

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

MMIC FILE COPY

86 11 10 98

AFIT/GSM/LSY/86S-4

A STUDY OF UPPER ERROR LIMITS
IN ACCOUNTING POPULATIONS

THESIS

G. Steven Bringle
Captain, USAF

AFIT/GSM/LSY/86S-4

DTIC
SELECTE
NOV 20 1986
B

Approved for public release; distribution unlimited

The contents of the document are technically accurate, and no sensitive items, detrimental ideas, or deleterious information are contained therein. Furthermore, the views expressed in the document are those of the author(s) and do not necessarily reflect the views of the School of Systems and Logistics, the Air University, the United States Air Force, or the Department of Defense.



Approved	✓
NTIS	
DTIC	
Un	
Just	
By	
Dist	
Avail	
Dist	
A-1	

AFIT/GSM/LSY/86S-4

A STUDY OF UPPER ERROR LIMITS
IN ACCOUNTING POPULATIONS

THESIS

Presented to the Faculty of the School of Systems and Logistics
of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Systems Management

G. Steven Bringle, B.S.

Captain, USAF

September 1986

Approved for public release; distribution unlimited

Acknowledgments

I wish to thank Jeffrey J. Phillips, my thesis advisor, for his encouragement and assistance throughout the research process. I would also like to express my gratitude to TSgt James Knight, whose help in obtaining the data for this research, made the final product possible. My final thanks is to my family, my wife, Valorie and children, Jaclyn, Gregory, and David, whose love and understanding have given me the impetus to see this research through.

G. Steven Bringle

Table of Contents

	Page
Acknowledgments	ii
List of Figures	v
List of Tables	vi
Abstract	viii
I. Introduction	1-1
General Problem	1-1
Specific Problem	1-3
Research Questions	1-4
Literature Review	1-5
Characteristics of Accounting Populations	1-6
Neter and Loebbecke Study . .	1-6
Johnson, Leitch, and Neter Study	1-7
Follow-up to Neter, Johnson, and Leitch Study	1-8
Ham, Losell, and Smieliauskas Study	1-8
Helton Study	1-9
Upper Error Limit Methods . .	1-10
Stringer Bound	1-10
Dollar Unit Sampling . .	1-11
Conclusion	1-13
II. Methodology	2-1
Data Collection	2-2
Computer Programs	2-2
The Models	2-4
Dollar Unit Sampling	2-4
Unmodified Stringer Bound . .	2-5
Modified Stringer Bound . .	2-7
Unmodified DUS-cell Bound . .	2-9
Modified DUS-cell Bound . .	2-10
III. Results and Analysis	3-1
Research Question One	3-2
Research Question Two	3-3
Research Question Three	3-4
Neter and Loebbecke Study . .	3-4

	Johnson, Leitch, and Neter	
	Study	3-5
	Ham, Losell, and Smieliauskas	
	Study	3-8
	Research Question Four	3-9
	Research Question Five	3-14
	Research Question Six	3-20
	Research Question Seven	3-22
IV.	Summary and Conclusions	4-1
	Summary	4-1
	Conclusions	4-3
	Suggestions for Further Research	4-4
Appendix A:	Computer Program 2	A-1
Appendix B:	Computer Program 3	B-1
Appendix C:	Computer Program 4	C-1
Appendix D:	Computer Program 5	D-1
Bibliography	Bib-1
Vita	V-1

List of Figures

Figure	Page
2.1. Example Unmodified Stringer Method (95% Confidence Level)	2-7
2.2. Example of Modified Stringer Bound (95% Confidence Level)	2-8
2.3. UEL Factor Computation (95% Confidence Level)	2-9
2.4. Example of Unmodified DUS-cell Method (95% Confidence Level)	2-10
2.5. Example of Modified DUS-cell Method (95% Confidence Level)	2-11
3.1. Average Percentage Change from the Nominal Confidence Level	3-21

List of Tables

Table	Page
3.1. Error Amount Intensity (EAI) Categories	3-1
3.2. Study Populations (Ps) by Error Amount Intensity (EAI)* Category for 13 Study Populations	3-2
3.3. Direction of Errors in the Population (Pop1)	3-3
3.4. Comparison of Major Characteristics of the Population Book Values	3-5
3.5. Comparison of Major Characteristics of the Population Book Values	3-5
3.6a. Coverage of Unmodified Bounds	3-10
3.6b. Coverage of Unmodified Bounds Error Amount Intensity (EAI) High (.098 - .148)	3-11
3.6c. Coverage of Unmodified Bounds Error Amount Intensity (EAI) Medium (.038 - .085)	3-11
3.6d. Coverage of Unmodified Bounds Error Amount Intensity (EAI) Low (.011 - .020)	3-12
3.7a. Coverage of Modified Bounds	3-12
3.7b. Coverage of Modified Bounds Error Amount Intensity (EAI) High (.098 - .148)	3-13
3.7c. Coverage of Modified Bounds Error Amount Intensity (EAI) Medium (.038 - .085)	3-13
3.7d. Coverage of Modified Bounds Error Amount Intensity (EAI) Low (.011 - .020)	3-14
3.8a. Relative Tightness for 95% Nominal Confidence Level Bounds	3-15

3.8b.	Relative Tightness for 95% Nominal Confidence Level Bounds Error Amount Intensity (EAI) High (.098 - .148)	3-16
3.8c.	Relative Tightness for 95% Nominal Confidence Level Bounds Error Amount Intensity (EAI) Medium (.038 - .085)	3-17
3.8d.	Relative Tightness for 95% Nominal Confidence Level Bounds Error Amount Intensity (EAI) Low (.011 - .020)	3-17
3.9a.	Relative Tightness for 85% Nominal Confidence Level Bounds	3-18
3.9b.	Relative Tightness for 85% Nominal Confidence Level Bounds Error Amount Intensity (EAI) High (.098 - .148)	3-19
3.9c.	Relative Tightness for 85% Nominal Confidence Level Bounds Error Amount Intensity (EAI) Medium (.038 - .085)	3-19
3.9d.	Relative Tightness for 85% Nominal Confidence Level Bounds Error Amount Intensity (EAI) Low (.011 - .020)	3-20
3.10.	Descriptive Statistics of Book Values for the Accounting Population	3-23
3.11.	Comparison of the Coverage of Unmodified Bounds	3-24
3.12.	Comparison of the Coverage of Modified Bounds	3-25
3.13.	Comparison of the Relative Tightness for 95 Percent Nominal Confidence Level Bounds	3-25
3.14.	Comparison of the Relative Tightness for 85 Percent Nominal Confidence Level Bounds	3-26
3.15.	Comparison of the Average Percentage Change of the Actual Confidence Level from the Nominal Confidence Level .	3-27

ABSTRACT

The purpose of this research was to examine a new accounts payable accounting population, comparing it to other populations which have been studied, examine the validity of an upper error limit bound, and compare those results with the results of a previous study. The bound examined was the Leslie, Teitlebaum, and Anderson DUS-cell bound. This method was supposed to reduce the bound conservatism and produce actual confidence levels closer to the nominal confidence levels.

The analysis of the DUS-cell bound was accomplished by examining the robustness, the relative tightness, and the effect of error amount intensity on the coverage provided and the relative tightness of the bound. The analysis of the other areas was by comparison.

The results of the research indicate that statistical characteristics varied for different accounting populations. The analysis of the validity of the DUS-cell bound method indicate that it is robust at the 95 percent confidence level, but is not at the 85 percent confidence level. In both cases, the DUS-cell bound is tighter than the Stringer bound. The results also indicate that error amount intensity significantly affects the coverage and the

relative tightness provided by the DUS-cell bound.

Comparing the results to a previous study of the validity of the bound provides mixed results. For this reason, further research needs to be accomplished in this area.

A STUDY OF UPPER ERROR LIMITS IN ACCOUNTING POPULATIONS

I. Introduction

General Problem

The financial statements of a business are the means for communicating with those outside that organization, such as investors, potential investors, lenders, and regulatory agencies, about the financial standing of the business in question. The three areas presented in financial statements are the business' financial position, the results of the operations for the last business year or cycle, and the changes in financial position from the previous accounting period to the one currently being presented. These financial statements are required to conform with generally accepted accounting principles. The independent audit process is the means used for verifying that conformity.

The independent audit process is performed for the business by an audit firm chosen and hired by that organization. The audit firm's auditors are required to render statements about the accuracy of the business' financial statements when the audit is complete. They make their determination of this accuracy based upon reviewing the internal controls of the organization in question and empirical sampling of the accounting populations, such as accounts receivable, accounts payable, inventory, etc.,

within that organization. The size of most of these populations is such that a 100 percent examination is not a cost or time effective method of verification. For this reason, empirical sampling is done and predictions of the total population characteristics are made good upon these samples. Obviously, there is risk involved in any sampling process, and the auditor and the audit firm want to accept a minimal risk of error before rendering judgment.

The traditional means for minimizing risk of error has been through the use of upper error limit bounds by the auditor. However, this method of prediction sometimes results in very large confidence intervals for the error, possibly far greater than the nominal confidence level of the bound and greater than the error actually present in the population. This trend toward conservatism is due, in part, to little being known about the error characteristics of accounting populations (6:270). This conservative approach may also lead to erroneous rejections of the hypothesis that the book value, the amount shown on the supporting accounting records before verification, of the audited population is reasonably accurate (11:501).

Another problem encountered by auditors is low error rates in the accounting populations. If an auditor is using one of the classical sampling techniques, such as ratio or difference estimators, and the sample contains no errors, a frequent occurrence, the estimated standard deviation, will

then be zero. A result of this nature is meaningless to the auditor. Since classical techniques assume a normal distribution of error, even samples containing a few errors can result in the actual confidence coefficient being substantially different than the predicted value (10:5).

The lack of knowledge about accounting populations has resulted in some studies being performed. These include the Neter and Loebbecke study published in 1975 (10), the Johnson, Leitch, and Neter study published in 1981 (6:270-293) with the follow-up in 1985 (8:488-499), the Ham, Losell, and Smieliauskas study published in 1985 (4:387-406) and the Helton study in 1985 (5). These studies have focused on five types of accounting populations in the for profit sector of the economy, accounts receivable, accounts payable, sales, inventory, and purchases. These studies have provided an initial look at the verification of the assumptions currently being made about the nature of accounting populations for the audit design process.

Specific Problem

A closer look at the true nature of accounting populations and the nature of the errors associated with each particular type of population could very well influence the audit design process. A clearer understanding of error distributions, the direction of the errors (understatements or overstatements), the magnitude of the errors, the error variability, the types of errors, and the shapes of the

error rate distribution associated with each type of accounting population could be useful. It would allow the testing of the error estimates currently being used by auditors or be cause for re-evaluation these estimators and even the development of new techniques which will better satisfy the auditor's desire to minimize the risk of error when rendering statements about the accuracy of a business' financial statements.

Another twist may be added when studying accounting populations. Both for profit and not for profit organizations exist in the economy. Do differences in the nature of accounting populations exist due to this fact? Through study this question may be verified.

Research Questions

The following research questions have been developed to accomplish an analysis of a specific accounting population chosen for study.

1. An auditor might expect to find understatement errors in the majority for an accounts payable population. Is this consistent with the population being studied?
2. What kind of statistical distribution best represents the population of errors being analyzed?
3. How does this population compare to previously studied populations of the same type? Of different types?

4. Do the Stringer, modified Stringer, DUS-cell, and modified DUS-cell bounds have an actual confidence level higher than the nominal confidence level being tested? In other words, are the bounds robust?
5. Does the DUS-cell bound method yield tighter confidence intervals than the Stringer bound? The purpose of this question is to determine which bounding method better reduces the potential for rejecting populations not materially in error.
6. Is there an effect on robustness of the modified or unmodified DUS-cell bound as the confidence level changes? The purpose of this question is to determine if the bound performs differently at different levels of confidence. If so, this would affect the amount of risk that the auditor would be accepting.
7. Are the results from the Helton study further verified by the new population from ASD?

Literature Review

This literature review chronologically presents information about studies of accounting populations and their results. These studies deal with the nature of the populations and the characteristics of the errors associated with those populations. Furthermore, this literature review will present some major studies that have examined methods of determining upper error limits.

Characteristics of Accounting Populations. Studies of accounting populations began because of the lack of knowledge about the sampling behavior of major statistical estimators and error patterns (10:5). These studies began with research published by Neter and Loebbecke in 1975 for the American Institute of Certified Public Accountants (10). Other studies have followed, including further work in the area by Neter with Johnson and Leitch published in 1981 (6:270-293). A follow-up to this work was published in 1985 (8:488-495). Another study, by Ham, Losell and Smieliauskas, has analyzed more actual accounting populations in detail (4:387-406).

Neter and Loebbecke Study. The Neter and Loebbecke study included analysis of four populations, three accounts receivable populations and one inventory population, with different error rates and error patterns. Initially, each population was analyzed to determine the characteristics of its book values, the nature of its error pattern, and the characteristics of the audit values for the study populations created with different error rates (10:11). By using actual error patterns to generate several study populations, it was possible to study the behavior of statistical estimators using different error rates for each error pattern (10:6-7). Each study population was then evaluated using three sampling techniques, simple random sampling, stratified random sampling and dollar unit

sampling, and several different statistical estimators. The purpose of this analysis was to determine the precision of the estimator and the reliability of the nominal confidence coefficient assuming a normal distribution for the population (10:7-9).

The findings suggested that an optimal statistical procedure under all circumstances for both characteristics did not exist. However, a reasonably effective procedure existed for each particular circumstance. The implication is that the auditor must be familiar with a variety of statistical sampling procedures and for what circumstance each procedure is most effective and appropriate (10:127-139).

Johnson, Leitch, and Neter Study. This study looked at the characteristics of errors in two types of accounting populations, accounts receivable and inventory. The study analyzed the error rates in each of the 55 accounts receivable populations and the 26 inventory populations. Of interest in this analysis was the balance of overstatement and understatement errors in each population (6:272-281). Also analyzed were the distribution of error amounts and error taints and the relationship between error amounts and book values (6:281-290).

The implications of this study are based upon the sample being taken from one Certified Public Accounting (CPA) firm with all populations based upon audits of larger

clients. The findings for the characteristics and shapes of error distributions for both types of populations were of interest, but could not necessarily be extrapolated for all CPA firms, nor for all sizes of client (6:291-292). The results again show that care must be taken when selecting statistical sampling procedures for a particular kind of population.

Follow-up to Neter, Johnson, and Leitch Study. This follow-up to the 1981 study mentioned above was initiated to consider the distribution of dollar-unit taints, the relation between the size of line-item taint and book value, and dollar-unit error rates. This further analysis was thought useful for those interested in dollar-unit sampling as an audit technique (8:488). The previous analysis had focused on information of use to those interested in applying one of the classical statistical sampling techniques. A comparison of the line-item approach and the dollar-unit approach was the result of the study.

The implications of this follow-up were again constrained by the sample on one CPA firm and the populations based upon audits of large clients. The results did re-emphasize the need for care in the selection of sampling procedure for the particular circumstance. Also re-emphasized was the need for further study.

Ham, Losell, and Smieliauskas Study. This study was an analysis of five accounting categories, their

characteristics and the associated error characteristics. Previously, studies had been limited to accounts receivable and inventory populations, but this study also included three others, as well as new populations of those types. The additional categories were populations for accounts payable, purchases, and sales. Another difference in this study was that the populations were derived from audits provided by a new CPA firm (4:387).

This study goes into detailed description of the error distributions of each population with regard to its shape and variability. Other factors are also discussed. Additionally, four error rates are defined and applied to each of the five populations. Then, their error distributions are presented and analyzed (4:395-401). Finally, four environmental factors are investigated for their effects on the four defined error rates (4:401-403).

For this study, more comparisons between accounting categories were possible. The results were similar to the previous works in this area. The underlying result was that the statistical sampling process chosen has its basis in the type of accounting population being tested. Consequently, the statistical sampling procedure chosen must be appropriate for that particular circumstance.

Helton Study. This study was the analysis of the Leslie, Teitlebaum, and Anderson DUS-cell method of upper error limit bounding. The DUS-cell method is supposed to

reduce bound conservatism while producing actual confidence levels closer to nominal confidence levels than those of the very conservative Stringer bound (5:2-1 -- 2-4).

The analysis focused on an examination of bound robustness, the bound relative tightness, and the effects of four factors, error rates, error clustering, mean taint, and error amount intensity, on the coverage and relative tightness of the DUS-cell bound as compared to the Stringer bound. The results of the study indicated that the DUS-cell bound did not perform to expectations. However, indications were that selective use of the bound might provide useful results. The most significant area of interest was how the error amount intensity affected coverage and relative tightness of the DUS-cell bound. This study also points to the need for further investigations before final determination can be made (5).

Upper Error Limit Methods.

Stringer Bound. Kenneth W. Stringer introduced the Stringer bound method of predicting the upper error limit in 1963. His motivation was that classical sampling techniques included an inherent danger for the auditor, if no errors were found through the sampling process, the estimated standard error would be zero, very possibly a meaningless result (9:77). The Stringer bound method is felt to have two major drawbacks. First, there is no theoretical statistical basis supporting the confidence

level attributed to the procedure (9:78). However, auditors are satisfied that it works, so they use it. Second, the confidence interval produced by the method is thought much too conservative, producing upper error limits much greater than the true error of the population. Although these flaws exist, the Stringer bound continues to be used as a benchmark for comparison with other methods for the prediction of upper error limits.

Dollar Unit Sampling. Dollar-unit sampling (DUS) is a modified version of attribute sampling. It was developed in an attempt to compensate for the suspected low error rates inherent in accounting populations. The unique feature of DUS is that the accounting population is viewed as individual dollar units rather than book values. When DUS is applied, a stratified sample is an immediate result because larger dollar units have a greater probability of random selection than smaller dollar units. In the application of DUS, three assumptions must be made. The first two assumptions are necessary because DUS uses the Poisson probability distribution to evaluate the sample. The first assumption is that the real error rate for the chosen population is small, less than ten percent. The second is that the population should be large, greater than 2,000 dollar units. The last assumption is not required, but allows for simplification. That assumption is that the amount of error in any individual item in the chosen

population cannot exceed 100 percent. Evaluation using DUS may be done in terms of error rate, dollar amount of error, or both (2:31).

Using DUS provides the auditor with a number of advantages:

1. By stratifying the sample, DUS increases the possibility of detecting large errors in large dollar and infrequent transactions.
2. DUS can be used for both variable and attribute sampling, simultaneously.
3. For attribute sampling, conclusions may be made in both dollar amounts and error rates.
4. When overstatement errors are expected and variable sampling will be employed, DUS may always be used.
5. Normal distribution is not a necessary assumption for the population of interest.
6. DUS is an acceptable procedure, meeting the objectives of SAS 39, issued by the AICPA.
7. It is easily used and only requires a single Poisson probability distribution table to evaluate sampling results (2:31).

Potential uses for DUS include account receivable valuation and authorizations, expense account authorizations, approval and amounts, valuation of physical inventories, and cash disbursement authorizations. There are only four relevant parameters required to implement the use of DUS. They are the book value of the population, the number of physical units or line items in the population, the dollar amount the auditor has set for materiality, and the confidence level the auditor is trying to establish concerning the upper error limit of the evaluation (2:31-32).

DUS, like the Stringer bound, contains two major flaws.

There is no apparent theoretical statistical basis for DUS, but it is widely used because its results are broadly correct (3:126). The DUS method is also felt to produce results which are much too conservative. In an attempt to narrow the gap between actual and nominal confidence levels, "Anderson and Teitlebaum [1973] proposed the use of dollar unit sampling with the Stringer bound" (9:78).

Conclusion

Auditors are faced with a complex task when going through the audit design process. The choice of appropriate sampling techniques is critical to the success of the audit in providing a realistic view of a company through its financial statements. The studies presented have all emphasized the need for further research work in the examination of accounting population characteristics and their associated errors. By so doing, new methods of audit may be developed, if required. The result of this research should be a more accurate and reliable audit decision on the part of the auditor. This in turn leads to a more accurate and reliable picture of the company through its financial statements.

II. Methodology

The research methodology used consisted of seven steps. The first of these steps was the collection of the data to be analyzed to satisfy the research objectives. The second step was the statistical analysis of that data using SPSSx, Release 2.1 for the VAX UNIX system (12;13). The third step was the modification of a computer program (1:271) to sort the data in ascending numeric order (Appendix A). The fourth step was the modification of a computer program (5:A-1 -- A-3) to randomly select line items from the data collected which would subsequently be seeded with error (Appendix B). The fifth step was the modification of a computer program (5:B-1 -- B-22) to seed the line items selected in the preceding step with error, create 13 study populations, and to then simulate the DUS-cell bound method (Appendix C). The sixth step was the use of a computer program (5:C-1 -- C-13) to compute the coverage and relative tightness for the study populations generated in the preceding program (Appendix D). The seventh and final step consisted of further statistical analysis of the population through the results of the computer programs of steps four through six and a comparison to the results found by Michael W. Helton in his 1985 AFIT thesis (5), A Validation of an Accounting Upper Error Limit Bound.

Data Collection

The accounting population used to satisfy the research objectives was one obtained from the Travel Accounting Branch of Aeronautical Systems Division (ASD), Wright-Patterson Air Force Base, Ohio. The population is comprised of 2986 accounts payable from ASD's temporary duty accounts. The book values for the population range from \$1.85 to \$8,110.66. The data received also included audit values for a random sample of approximately ten percent of the population, 311 audit values. The audit values range from \$1.85 to \$2,743.86.

Computer Programs

The first computer program was written for the purpose of determining some of the major statistical characteristics of the book value population. This was accomplished through the use of SPSSx Release 2.1 for the VAX UNIX system (12;13). Characteristics of the book value population calculated included the mean, standard deviation, variance, skewness, and kurtosis.

The second computer program was a modified version of an integer sort program (1:271), the program, as modified, is in Appendix A. The data needed to be sorted in ascending numeric order for the fourth computer program to be able to perform a strip off subroutine required for the later analyses.

The next modified computer program provided the first step in the sample generation (5:A-1 -- A-3). This program randomly selects line items to be subsequently seeded with error. Inherent in this process, was a determination of the median dollar amount and its line item number. By doing this, the population was split into a high and a low category. This was done because there exists an equal probability of error occurring in each range of line items. An error rate of 80 percent of the line items in each category was arbitrarily selected for the error seeding process.

The fourth computer program was modified to accommodate only one population (5:B-1 -- B-22). The program was used to generate 13 study populations to be evaluated through an error allocation in three dimensions; error rates, error clustering, and mean taints. Three categories of error rates were simulated, a high level of line item error rate, a medium level, and a low level. Four error rates were arbitrarily selected to represent these categories at .50, .30, .15, and .01. The mid range category was doubly represented by the .30 and .15 error rates. The study populations were then sampled 500 times with a sample width of 200 dollar units. The sampling technique, known as Dollar Unit Sampling (DUS), was used. In using this method, all items of a dollar amount greater than the average sampling interval were deleted from the sampling routine.

In so doing, it was assumed that those items would be audited on a 100% basis while the remainder would be tested using the DUS sampling basis.

The fifth computer program was designed to compute the coverage and relative tightness for the study populations generated in the preceding program (5:C-1 -- C-13). This was accomplished through the calculation of an upper error limit bound (UEL) for each DUS sample and for each of the four bound types, unmodified Stringer, modified Stringer, unmodified DUS-cell, and modified DUS-cell. Two levels of confidence were measured against each of these bounds, one at 95 percent and the other at 85 percent. Therefore, eight UELs were calculated for each study population. Analysis of the UELs was based upon the coverage of each of these bounds and how the coverage varied with respect to the error amount intensity. Also examined was the relative tightness of each of the other three bounds with the unmodified Stringer bound at the same confidence level. Further analysis also looked at how the relative tightness varied with respect to the error amount intensity.

The Models

Dollar Unit Sampling. The dollar unit sampling (DUS) technique is a method of sampling in which each dollar-unit of the population has an equal probability of being sampled. This sampling approach divides the population of accounts into an arbitrarily chosen number of "equal cells", and then

one dollar-unit from each cell is randomly selected (7:103). For example, consider an accounting population having a total book value of \$100,000. If the auditor desires to have 200 sample elements drawn, then the "equal cells" or sampling interval would be \$500 ($\$100,000/200$). Once the sampling interval is determined, one dollar-unit is randomly selected from each cell and the line item number containing each of those dollar-units are audited. An additional component which must be considered during this process is the taint of the reported book value. The taint is defined as "the ratio of error amount (whether overstatement or understatement) to the reported book value of the physical unit (e.g., invoice, account balance, etc.) in which it occurs" (7:122). In other words, the taint is calculated by subtracting the audit value of a line item from the reported book value and then dividing by the reported book value. This results in taints being expressed as real numbers between the values of zero and one, including those outer limits. Employing this definition has the effect of distributing the error uniformly throughout the sample interval. The taint is computed during the random sample process.

Unmodified Stringer Bound. The Stringer bound method was first introduced by Kenneth W. Stringer in 1963. This method is based upon the assumption that errors in accounting populations follow a Poisson approximation to the

Binomial distribution. The application of this assumption in calculating an upper error limit (UEL) bound requires the use of three factors; basic precision (BP), most likely error (MLE), and precision gap widening (PGW). Basic precision is a factor of UEL based upon the number of errors found through the random sampling process and a particular confidence level. Due to an auditor's tendency toward conservatism, the BP factor is always greater than zero. In other words, the BP factor, when no errors are found, is the floor the auditor would intuitively expect for the accounting population and each sample error "will raise the final upper error limit higher than this floor" (7:127). For the purpose of this study, two confidence levels have been selected one at 95 percent and the other at 85 percent. The BP values associated with these confidence levels are always 3.00 and 1.90, respectively. The most likely error factor is "simply a projection of the sample error rate found" (7:127). In other words, the MLE will equal the number of errors found during the sampling process. The precision gap widening factor is the amount the basic precision changes as the result of finding a sample error (7:127).

To find the upper error limit for the unmodified Stringer bound requires the three factors, taintings in descending order, and the average sampling interval. An example is provided in Figure 2.1 for a 95 percent

confidence level, an average sampling interval of \$500, and a sample of four errors with taints as represented in the figure. The process requires three calculations using this information. The first calculation is solving for the MLE column, E, which is a cumulative total of column B. The second is the calculation of a cumulative total of the product of column B and column C. The last is the calculation of a cumulative total of the sum of column D, column E, and column F. This last calculation results in the stage UEL. The stage UEL is then multiplied by the

A Error Stage	B Taint- ings	C PGW Factor	D BP	E MLE (+B)	F PGW (B*C)	G Stage UEL (D+E+F)
0	-		3.0	-	-	3.000
1	.9	.75		.90	.675	4.575
2	.8	.55		1.70	1.115	5.815
3	.5	.46		2.20	1.345	6.545
4	.1	.40		2.30	1.385	6.685

(5:2-6)

Figure 2.1. Example Unmodified Stringer Method
(95% Confidence Level)

$$\begin{aligned}
 \text{UEL} &= \text{Stage UEL} * \text{Average Sampling Interval} & (1) \\
 &= 6.685 * \$500 \\
 &= \$3343
 \end{aligned}$$

average sampling interval, as shown in equation (1) with the estimated upper error limit as its result.

Modified Stringer Bound. For the modified Stringer bound method, the same method of calculation is used for the

stage UEL. The difference in the calculation of the modified Stringer UEL is that the stage UEL is projected over the average dollar amount not sampled and then the total error amount is added. An example of this method, using a sample of 200 drawn from a population of \$100,000, with a total book value for the sample of \$15,000, and with four errors

Error	Taint	Audit Value	Book Value	Error Amount (C - D)
1	.5	\$ 120	\$ 80	\$ 40
2	.9	380	200	180
3	.1	1100	1000	100
4	.8	225	125	100
Total			\$1265	\$ 420

(5:2-7)

Figure 2.2. Example of Modified Stringer Bound (95% Confidence Level)

present having taintings of .50, .90, .10, and .80 and book values of \$80, \$200, \$1000, and \$125, respectively, is shown in Figure 2.2. From this information, the UEL may be calculated by using the following formula:

$$\begin{aligned}
 \text{UEL} &= (\text{Ty} - \text{sampbv})/n * \text{stage UEL} + \text{samper} & (2) \\
 &= (\$100,000 - \$15,000)/200 * 6.685 + \$420 \\
 &= \$3261
 \end{aligned}$$

where:

Ty = total book value of the population
 sampbv = total book value of the sample
 n = sample size
 stage UEL = as calculated from the unmodified Stringer method
 samper = total amount of error in the sample

Unmodified DUS-cell Bound. The unmodified DUS-cell bound method uses the dollar unit sampling technique described previously. In this method a UEL factor is calculated using the same three components as were used in the unmodified Stringer bound method. The treatment is, however, somewhat different. The UEL factor is calculated by summing the BP, the total MLE, and the total PGW as shown in Figure 2.3.

Error Stage	BP	MLE	PGW	UEL
0	3.00	-	-	3.00
1		1.00	.75	4.75
2		1.00	.55	6.30
3		1.00	.46	7.76
4		1.00	.40	9.16
	3.00	4.00	2.16	

(7:125)

Figure 2.3. UEL Factor Computation
(95% Confidence Level)

Leslie, Teitlebaum, and Anderson have developed tables which provide the UEL and lower error limit (LEL) factor for a corresponding number of errors, and a particular confidence level.

Once the UEL factors and taints are determined, two more calculations are necessary. The first of these is the "load and spread" calculation. To make this calculation "take the worst error pattern of the preceding stage modified by loading one cell full of the most recent

tainting" (7:141). The other calculation is the "simple spread," which is computed by multiplying the UEL factor by the cumulative average taint from error stage 1 on. The "simple spread" value at error stage 0 is the UEL factor. For example, if a sample of 200 is drawn from a population with a reported book value of \$100,000, the sampling interval would be \$500. If the sampling yields four errors with respective taintings of .90, .80, .50, and .10, then the upper error limit at a 95 percent confidence level would be:

A Error Stage	B UEL Factor	C Taint- ings	D Cum Avg Tainting	E UEL of Prev. Stage (H)	F Load and Spread (E+C)	G Simple Spread (B*D)	H Stage UEL Max (F,G)
0	3.00	1.00	-	-	-	3.000	3.000
1	4.75	.90	.90	3.000	3.900	4.275	4.275
2	6.30	.80	.85	4.275	5.075	5.355	5.355
3	7.76	.50	.73	5.355	5.855	5.665	5.855
4	9.16	.10	.58	5.855	5.955	5.313	5.955

(7:142)

Figure 2.4. Example of Unmodified DUS-cell Method
(95% Confidence Level)

$$\begin{aligned}
 \text{UEL} &= \text{Stage UEL} * \text{Average Sampling Interval} & (1) \\
 &= 5.955 * \$500 \\
 &= \$2978
 \end{aligned}$$

Modified DUS-cell Bound. The modified DUS-cell bound method calculates the stage UEL in the manner used in the unmodified DUS-cell bound method. When calculating the UEL, however, the modified DUS-cell bound method follows the method used in computing the modified Stringer bound. The

stage UEL is again projected over the average dollar amount not sampled and then the total amount of error in the sample is added to that result. For example, assume, as in the unmodified DUS-cell bound method, that a sample of 200 is taken from a population with a reported total book value of \$100,000 and that four errors with taintings of .50, .90, .10, and .80 are found in the sampling process. These taintings are associated with reported book values of \$80, \$200, \$1000, and \$125, respectively. Using the modified DUS-cell bound method would result in the following UEL calculation (Note: The calculation of the error amount presented in Figure 2.5 is the same as that portrayed in Figure 2.2 for the modified Stringer bound):

Error	Taint	Audit Value	Book Value	Error Amount (C - D)
1	.5	\$ 120	\$ 80	\$ 40
2	.9	380	200	180
3	.1	1100	1000	100
4	.8	225	125	100
Total			\$1265	\$ 420

Figure 2.5. Example of Modified DUS-cell Method
(95% Confidence Level)

$$\begin{aligned}
 \text{UEL} &= (\text{Ty} - \text{sampbv})/n * \text{stage UEL} + \text{samper} & (2) \\
 &= (\$100,000 - \$15,000)/200 * 5.955 + \$420 \\
 &= \$2951
 \end{aligned}$$

where:

Ty = total book value of the population
 sampbv = total book value of the sample
 n = sample size

stage UEL = as calculated from the unmodified DUS-cell
method
samper = total amount of error in the sample

III. Results and Analysis

This chapter contains the analysis of the study. The chapter is divided into seven major headings, one for each research question. Throughout the analysis, the data is examined with the error amount intensity (EAI), grouping results into three categories, low, medium, and high. Table 3.1 shows the ranges of the groupings and Table 3.2 ranks the error amount intensity from low to high for each category. The subdivision points of the EAI are the same as were used in the Helton study.

Table 3.1
Error Amount Intensity (EAI) Categories

Category	EAI* Range	Number of Study Populations
Low	.011 - .020	3
Medium	.038 - .085	4
High	.098 - .148	6
Total		13
* Study population total dollar error amount as a proportion of total book value		

The error amount intensity is a population characteristic obtained by dividing the total population dollar error by the total population book value. The analysis addresses the study populations as a whole and then looks at the performance within these three categories. This analysis is done for the coverage and relative tightness for each bound.

 Table 3.2
 Study Populations (Ps) by Error Amount
 Intensity (EAI)* Category for 13
 Study Populations

Low (3)		Medium (4)		High (6)	
Ps	EAI	Ps	EAI	Ps	EAI
12	.0105	9	.0381	7	.0975
13	.0105	10	.0423	2	.1033
11	.0197	8	.0597	5	.1101
		6	.0853	4	.1130
				1	.1279
				3	.1484

 * Study population error as a proportion of total book
 value

Research Question One

The first research question addressed the types of errors, overstatements and understatements, found in the population. Since the population was made up of accounts payable, auditors might expect understatement errors to be in the majority. For this population, this was not the case. Of the sample of 311 book values, 22 were found in error. Of these 22, 9 were understatement errors (41 percent) and 13 were overstatement errors as shown in Table 3.3.

 Table 3.3
 Direction of Errors in the Population (Popl)

Overstatements		Understatements	
Line Item	Amount of Error	Line Item	Amount of Error
114	15.30	205	12.00
154	12.30	813	7.50
215	1.36	1374	12.50
260	9.00	1404	25.00
846	22.50	1783	6.90
1384	6.00	1930	7.00
1604	79.50	2144	7.50
1949	12.00	2574	15.00
2240	8.00	2744	25.00
2399	100.00		
2409	4.00		
2539	10.00		
2883	25.00		

Research Question Two

The second research question addressed the type of distribution which was best represented by the errors found in sampling the population. The number of errors found was 22 out of a sample of 311, about seven (7) percent. Because the number of errors found was so small, it would be meaningless to try and decide which distribution these errors best represented. To get a more meaningful representation would require either a larger population with the same error rate and sample size or a larger error rate. The error amount intensity (EAI) associated with this population was .000554 (\$423.36/\$763,931.19). This population characteristic is of interest because the results of the simulation done for research questions four through

seven indicate that the unmodified DUS-cell bound method works best in the low range of EAI, which is where the actual EAI for this population falls. The best coverage was provided by the bound in this range of EAI.

Research Question Three

The third research question addressed a comparison of the results of this study with some of the other studies that have been done. Of the five studies mentioned in the literature review, this question will address three of those studies. Research question seven addresses one of the other two, the Helton study. The last study was the follow-up to the Johnson, Leitch, and Neter study (8). It will not be addressed because it focuses on the error taint and that has not been addressed in this study.

Neter and Loebbecke Study. The Neter and Loebbecke study involved four populations and the creation of five study populations for each. The five study populations were based upon error percentages of .5, 1, 5, 10, and 30. For comparison, each major book value statistical characteristics are compared in Tables 3.4a and 3.4b. The first population (NPop1), the third population (NPop3), and the fourth population (NPop4) were all accounts receivable accounts. The second population (NPop2) was an inventory account. The population used for this study had an error rate of about 7 percent which is most nearly associated with 5 percent, therefore the focus will be on the study populations created with that error percentage.

Table 3.4
Comparison of Major Characteristics of the
Population Book Values

	Pop1	NPop1	NPop2
Total Book Value	\$763,931.19	\$ 379,131.00	\$3,486,530.00
Mean	\$ 255.84	\$ 45.63	\$ 636.00
Std deviation	\$ 364.23	\$ 132.61	\$ 1,155.99
Skewness	8.5	22.0	3.5
Kurtosis	125.9	906.4	15.2
Maximum	\$ 8,110.66	\$ 6,869.70	\$ 9,989.00
Minimum	\$ 1.85	\$.50	\$ 1.00

Table 3.5
Comparison of Major Characteristics of the
Population Book Values

	Pop1	NPop3	NPop4
Total Book Value	\$763,931.19	\$13,671,500.00	\$7,502,957.00
Mean	\$ 255.84	\$ 1,945.84	\$ 1,860.39
Std deviation	\$ 364.23	\$ 7,021.61	\$ 3,865.13
Skewness	8.5	7.9	3.2
Kurtosis	125.9	78.1	11.4
Maximum	\$ 8,110.66	\$ 98,162.70	\$ 24,928.60
Minimum	\$ 1.85	\$.10	\$.10

The population in this study does not have book value characteristics which are similar to any of the Neter and Loebbecke populations.

Johnson, Leitch, and Neter Study. The Johnson, Leitch, and Neter study examined 55 accounts receivable and 26 inventory populations. Their study first examined the distributions of the book value balances for the populations. The book value balance for the population from this study (\$763,931.19) would have fallen into their lowest

category, between \$.25 million and \$.99 million (6:273). Their study then examined error rates. There was a great variability in error rates for both types of populations, accounts receivable and inventories, and "the error rates for inventory tend to be substantially higher than those for accounts receivable" (6:274). The error rate for the population from this study, seven percent, would have fallen into the next to the lowest error rate category for accounts receivable accounts with 21.9 percent of the total accounts audited. The same position was true in comparison to the inventory categories, and that category was made up of 19.3 percent of the populations audited (6:274). In both cases, these categories had the third greatest frequency of occurrence. This population's error rate was greater than the median error rate for accounts receivable accounts and less than the median error rate for inventory accounts (6:274-275). For the relationship of error rate to mean line item size, this study's mean line item size, \$ 255.84, would fall into the second category for accounts receivable. The median error rate for that category is 1.2 percent compared to the seven (7) percent in this study. For the inventory accounts, this study's result would fall somewhere between the first two categories. The corresponding median error rates are 8.3 percent and 15.4 percent, respectively, more in line with the result of this study (6:275-276).

The Johnson, Leitch, and Neter study looked for a relationship between the size of the individual line items and its susceptibility to error. The information from this study would be in line with what they found, accounts with lower mean book values per line item tend to have lower error rates. This study tended to support this, having a relatively low mean book value per line item, \$ 255.84, and a low error rate, seven (7) percent.

In comparison to the Johnson, Leitch, and Neter study, this study's frequency of overstatement and understatement errors tend to be like those found for inventory accounts. The overstatement rate in this study was 59 percent. This indicates a somewhat reasonable balance between the two error types. This result would fall within the range of categories which include 61.5 percent of the inventory audits. In comparison to the accounts receivable audits, that range includes only 1.8 percent of those audits, whereas 90 percent of the audits fell into the 90 percent to 100 percent overstatement rate (6:279-280).

This comparison indicates that this accounts payable population does not strictly follow the results of either the accounts receivable or inventory populations as studied by Johnson, Leitch, and Neter. Since this study examined only one small population, the results could not be considered conclusive. This initial look indicates that an audit design for and accounts payable population should be

treated differently than one for accounts receivable or inventory.

Ham, Losell, and Smieliauskas Study. The Ham, Losell, and Smieliauskas study examined error characteristics for 20 companies in five accounting categories. The five categories were accounts receivable, accounts payable, inventory, purchases, and sales. In comparing the results of this study to the results of their study for the accounts payable population, some contradictions were found. First, their accounts payable population errors tended to be in the direction of understatements, whereas the opposite was true for this study (4:390-391). The results of this study were more like the accounts receivable and sales results of their study. Second, in their study, "the accounts payable category has the highest number of errors in relation to the number of items tested" (4:403). In this study, only 22 errors were found for 311 line items tested, a low error rate incidence.

The error taint is "defined as the error amount of a line item in error divided by the book value of the line item and is fundamental to the dollar unit sampling technique" (4:398). The taint found in this study was similar to that found in the Ham, Losell, and Smieliauskas study, ranging from -.33 to .28 in this study. The majority of taints in their study also fell into this range, about 82 percent of the taints (4:400).

The findings of this study support those found in the Ham, Losell, and Smieliauskas study. In the two areas of conflict, the size of the sample and the low error rate of this study could make these results less significant.

Research Question Four

The fourth research question addressed the robustness of each of the bounds. Robustness refers to whether or not the coverage of the bounds met or exceeded the nominal level of confidence being tested. If the coverage meets or exceeds the nominal confidence level, it is said to be robust. If the overall coverage of a bound is robust, then the coverage in each of the categories is also robust. If the overall coverage of a bound is not robust, the different categories were analyzed to see if the different categories were robust.

The results for the coverage of the unmodified bounds are shown in Tables 3.6a, 3.6b, 3.6c, and 3.6d. The unmodified Stringer bounds were found to be robust at both nominal levels (S1, S3). The unmodified DUS-cell bound at the 95 percent confidence level (C1) was also robust. The unmodified DUS-cell bound at the 85 percent confidence level (C3) was not robust. The study populations which fell into the medium and low categories for that bound were robust, while the coverage at the high category was not robust. There was only one failure in the high category, and it missed qualifying by less than three percent.

Table 3.6a
Coverage of Unmodified Bounds

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S1	S3	C1	C3
1	.9920	.9400	.9500	.8220
2	1.0000	.9780	.9660	.9140
3	1.0000	.9940	.9920	.9260
4	1.0000	.9880	.9880	.9300
5	1.0000	1.0000	1.0000	.9300
6	1.0000	.9840	.9700	.9240
7	1.0000	.9940	.9860	.8980
8	1.0000	.9820	.9760	.8660
9	1.0000	1.0000	1.0000	.9460
10	1.0000	1.0000	1.0000	.9180
11	1.0000	.9980	1.0000	.9380
12	1.0000	1.0000	1.0000	1.0000
13	1.0000	1.0000	1.0000	1.0000
Mean Coverage	.9994	.9883	.9868	.9240

The results of the coverage of the modified bounds are shown in Tables 3.7a, 3.7b, 3.7c, and 3.7d. The coverage of the modified bounds provided one robust bound, the modified Stringer bound at the 85 percent confidence level (S4). None of the other bounds were close to providing coverage. The study populations that fell into the high and medium categories for the other three bounds still did not provide coverage and were not robust. The study populations falling into the low category did provide coverage and were robust for all three of the other bounds, the modified Stringer at the 95 percent confidence level and the modified DUS-cell bound at both confidence levels.

Table 3.6b
Coverage of Unmodified Bounds
Error Amount Intensity (EAI)
High (.098 - .148)

Confidence Level	95%	85%	95%	85%
	Stringer Bound		DUS-cell bound	
Study Population	S1	S3	C1	C3
1	.9920	.9400	.9500	.8220
2	1.0000	.9780	.9660	.9140
3	1.0000	.9940	.9920	.9260
4	1.0000	.9880	.9880	.9300
5	1.0000	1.0000	1.0000	.9300
7	1.0000	.9940	.9860	.8980
Mean Coverage	.9987	.9823	.9803	.9033

Table 3.6c
Coverage of Unmodified Bounds
Error Amount Intensity (EAI)
Medium (.038 - .085)

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S1	S3	C1	C3
6	1.0000	.9840	.9700	.9240
8	1.0000	.9820	.9760	.8660
9	1.0000	1.0000	1.0000	.9460
10	1.0000	1.0000	1.0000	.9180
Mean Coverage	1.0000	.9915	.9865	.9135

Table 3.6d
Coverage of Unmodified Bounds
Error Amount Intensity (EAI)
Low (.011 - .020)

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S1	S3	C1	C3
11	1.0000	.9980	1.0000	.9380
12	1.0000	1.0000	1.0000	1.0000
13	1.0000	1.0000	1.0000	1.0000
Mean Coverage	1.0000	.9993	1.0000	.9793

Table 3.7a
Coverage of Modified Bounds

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S2	S4	C2	C4
1	.8940	.8740	.8940	.6940
2	.9320	.9600	.9320	.8560
3	.9440	.9400	.9440	.7820
4	.9760	.9780	.9760	.8200
5	.9200	.9180	.9200	.7280
6	.9000	.9080	.9000	.5900
7	.8860	.8800	.8860	.6960
8	.8720	.9080	.8720	.6200
9	.9780	.9780	.9780	.8100
10	.9380	.9580	.9380	.7640
11	1.0000	.9480	1.0000	.8620
12	1.0000	.9980	1.0000	.9860
13	1.0000	1.0000	1.0000	.9980
Mean Coverage	.9415	.9422	.9415	.7851

Table 3.7b
Coverage of Modified Bounds
Error Amount Intensity (EAI)
High (.098 - .148)

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S2	S4	C2	C4
1	.8940	.8740	.8940	.6940
2	.9320	.9600	.9320	.8560
3	.9440	.9400	.9440	.7820
4	.9760	.9780	.9760	.8200
5	.9200	.9180	.9200	.7280
7	.8860	.8800	.8860	.6960
Mean Coverage	.9253	.9250	.9253	.7627

Table 3.7c
Coverage of Modified Bounds
Error Amount Intensity (EAI)
Medium (.038 - .085)

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S2	S4	C2	C4
6	.9000	.9080	.9000	.5900
8	.8720	.9080	.8720	.6200
9	.9780	.9780	.9780	.8100
10	.9380	.9580	.9380	.7640
Mean Coverage	.9220	.9380	.9220	.6960

 Table 3.7d
 Coverage of Modified Bounds
 Error Amount Intensity (EAI)
 Low (.011 - .020)

Confidence Level	95%	85%	95%	85%
	Stringer bound		DUS-cell bound	
Study Population	S2	S4	C2	C4
11	1.0000	.9480	1.0000	.8620
12	1.0000	.9980	1.0000	.9860
13	1.0000	1.0000	1.0000	.9980
Mean Coverage	1.0000	.9820	1.0000	.9487

Research Question Five

The fifth research question concerned the relative tightness of the bounds. The question of tightness only becomes relevant if the bound coverage is robust. The objective of this question was to compare the Stringer bounds to the DUS-cell bounds and the unmodified Stringer bound to the modified Stringer bound. The purpose of this comparison was to determine whether both DUS-cell bounds and the modified bound would yield a tighter bound than the unmodified Stringer bound. The results from the calculations described below can be found in Tables 3.8a, 3.8b, 3.8c, 3.8d, 3.9a, 3.9b, 3.9c, and 3.9d.

To determine the relative tightness of the Stringer bounds to the DUS-cell bounds, a quotient was first calculated. Each Stringer bound at each nominal confidence level was divided by the corresponding DUS-cell bound and

nominal confidence level. Next, these results were summed over 500 replications of the first calculation for each study population. Finally, these sums were divided by the number of replications, 500.

 Table 3.8a
 Relative Tightness for 95% Nominal Confidence
 Level Bounds

Study Population	Unmodified		Modified	
	S1*	C1**	S2*	C2**
1	1.0000	1.0914	1.0631	1.0816
2	1.0000	1.1218	1.0584	1.1080
3	1.0000	1.0866	1.0696	1.0780
4	1.0000	1.1165	1.0666	1.1043
5	1.0000	1.1056	1.0923	1.0972
6	1.0000	1.1291	1.1040	1.1200
7	1.0000	1.1110	1.1024	1.1031
8	1.0000	1.1584	1.1093	1.1477
9	1.0000	1.1931	1.1204	1.1817
10	1.0000	1.1831	1.1195	1.1723
11	1.0000	1.1991	1.1350	1.1901
12	1.0000	1.1810	1.1431	1.1743
13	1.0000	1.1812	1.1431	1.1745

Mean Relative Tightness	1.0000	1.1429	1.1021	1.1333

* Stringer bound
 ** DUS-cell bound

To determine the relative tightness of the unmodified Stringer bound to the modified Stringer bound, another quotient was calculated. Each unmodified Stringer bound was divided by the corresponding modified Stringer bound. As before, these results were summed over 500 replications of the previous calculation for each study population. The final step was the division of these sums by the number of replications, 500.

Three general results can be expected from the above processes. First, the result could be equal to one. This result would indicate that the two bounds relative tightness are equal. Second, the result could be greater than one. This result would indicate that the bound being compared to the unmodified Stringer bound would provide a "tighter" bound. Third, the result could be less than one. This result would indicate that the unmodified Stringer bound provided the tighter bound.

In all cases of comparison to the unmodified Stringer by this study, the bound being compared provided the tighter bound. As stated before, these results are only relevant

 Table 3.8b
 Relative Tightness for 95% Nominal Confidence
 Level Bounds
 Error Amount Intensity (EAI)
 High (.098 - .148)

Study Population	Unmodified		Modified	
	S1*	C1**	S2*	C2**
1	1.0000	1.0914	1.0631	1.0606
2	1.0000	1.1218	1.0584	1.1080
3	1.0000	1.0866	1.0696	1.0780
4	1.0000	1.1165	1.0666	1.1043
5	1.0000	1.1056	1.0923	1.0972
7	1.0000	1.1110	1.1024	1.1031
Mean Relative Tightness	1.0000	1.1055	1.0754	1.0954

* Stringer bound

** DUS-cell bound

Table 3.8c
Relative Tightness for 95% Nominal Confidence
Level Bounds
Error Amount Intensity (EAI)
Medium (.038 - .085)

Study Population	Unmodified		Modified	
	S1*	C1**	S2*	C2**
6	1.0000	1.1291	1.1040	1.1200
8	1.0000	1.1584	1.1093	1.1477
9	1.0000	1.1931	1.1204	1.1817
10	1.0000	1.1831	1.1195	1.1723
Mean Relative Tightness	1.0000	1.1659	1.1133	1.1554

* Stringer bound
** DUS-cell bound

Table 3.8d
Relative Tightness for 95% Nominal Confidence
Level Bounds
Error Amount Intensity (EAI)
Low (.011 - .020)

Study Population	Unmodified		Modified	
	S1*	C1**	S2*	C2**
11	1.0000	1.1991	1.1350	1.1901
12	1.0000	1.1810	1.1431	1.1743
13	1.0000	1.1812	1.1431	1.1745
Mean Relative Tightness	1.0000	1.1871	1.1404	1.1796

* Stringer bound
** DUS-cell bound

 Table 3.9a
 Relative Tightness for 85% Nominal Confidence
 Level Bounds

Study Population	Unmodified		Modified	
	S3*	C3**	S4*	C4**
1	1.0000	1.0622	1.0520	1.0551
2	1.0000	1.0825	1.0463	1.0726
3	1.0000	1.0585	1.0599	1.0523
4	1.0000	1.0787	1.0557	1.0699
5	1.0000	1.0718	1.0834	1.0656
6	1.0000	1.0885	1.0952	1.0817
7	1.0000	1.0759	1.0939	1.0700
8	1.0000	1.1105	1.0994	1.1023
9	1.0000	1.1382	1.1102	1.1291
10	1.0000	1.1304	1.1097	1.1218
11	1.0000	1.1521	1.1247	1.1439
12	1.0000	1.1470	1.1323	1.1402
13	1.0000	1.1472	1.1324	1.1404
Mean				
Relative	1.0000	1.1033	1.0919	1.0958
Tightness				
* Stringer bound				
** DUS-cell bound				

when the bound providing the coverage is robust. This means that the tighter bound provided by the unmodified DUS-cell bound and the modified Stringer bound at the 95 percent confidence level is important. For those bounds that were not robust across all study populations, but were in particular categories, the tighter bound is important for the low and medium categories for the unmodified DUS-cell bounds at 85 percent confidence level and for the low category for the modified bounds at all nominal confidence levels.

Table 3.9b
Relative Tightness for 85% Nominal Confidence
Level Bounds
Error Amount Intensity (EAI)
High (.098 - .148)

Study Population	Unmodified		Modified	
	S3*	C3**	S4*	C4**
1	1.0000	1.0622	1.0520	1.0551
2	1.0000	1.0825	1.0463	1.0726
3	1.0000	1.0585	1.0599	1.0523
4	1.0000	1.0787	1.0557	1.0699
5	1.0000	1.0718	1.0834	1.0656
7	1.0000	1.0759	1.0939	1.0700
Mean Relative Tightness	1.0000	1.0716	1.0652	1.0643
* Stringer bound				
** DUS-cell bound				

Table 3.9c
Relative Tightness for 85% Nominal Confidence
Level Bounds
Error Amount Intensity (EAI)
Medium (.038 - .085)

Study Population	Unmodified		Modified	
	S3*	C3**	S4*	C4**
6	1.0000	1.0885	1.0952	1.0817
8	1.0000	1.1105	1.0994	1.1023
9	1.0000	1.1382	1.1102	1.1291
10	1.0000	1.1304	1.1097	1.1218
Mean Relative Tightness	1.0000	1.1169	1.1036	1.1087
* Stringer bound				
** DUS-cell bound				

Table 3.9d
Relative Tightness for 85% Nominal Confidence
Level Bounds
Error Amount Intensity (EAI)
Low (.011 ~ .020)

Study Population	Unmodified		Modified	
	S3*	C3**	S4*	C4**
11	1.0000	1.1521	1.1247	1.1439
12	1.0000	1.1470	1.1323	1.1402
13	1.0000	1.1472	1.1324	1.1404
Mean Relative Tightness	1.0000	1.1488	1.1298	1.1415
* Stringer bound				
** DUS-cell bound				

Research Question Six

The sixth research question concerned the effect the choice of nominal confidence levels had on the robustness of the DUS-cell bounds. The reason for this examination is that the DUS-cell bounds may perform differently under assumptions of greater risk. A calculation was made to determine the average percentage change from the nominal confidence level for each DUS-cell bound. These calculations were accomplished as indicated in Figure 3.1.

The results indicate that, on the average, at a 95 percent nominal confidence level, the unmodified DUS-cell bound (C1) produced an actual coverage greater than is required by approximately 74 percent. In other words, for the 95 percent nominal confidence level the unmodified DUS-

A DUS-cell Bound	B Actual Confidence Level	C Nominal Confidence Level	D Differ- ence (B-C)	E Possible Increase (1.00-C)	F Avg % Change (D/E)
C1	.9868	.95	.0368	.05	.74
C2	.9415	.95	(.0085)	.05	(.17)
C3	.9240	.85	.0740	.15	.49
C4	.7851	.85	(.0649)	.15	(.43)

Figure 3.1. Average Percentage Change from
the Nominal Confidence Level

cell bound actually allows the auditor to assume less actual risk of error, thereby improving the probability of rendering an appropriate statement about the accuracy of the business' financial statements. The results for the unmodified coverage at the 85 percent nominal confidence level (C3) also produced actual coverage greater than required. Comparing the additional coverage provided by the unmodified DUS-cell bound at each nominal level of confidence indicates that there is a decrease in the benefit of the bound as the auditor assumes more risk. This benefit decreased by approximately one-third when the nominal confidence level changed from 95 percent to 85 percent. Though this decrease occurs, the benefit at the 85 percent level of confidence is still significant.

However, the modified DUS-cell bound produced dissimilar results. For both nominal confidence levels, the modified DUS-cell bound provides less coverage than is

required. In other words, the auditor would be assuming more actual risk than was desired by using the modified DUS-cell bound, thereby increasing the probability of rendering an inappropriate statement about the accuracy of the business' financial statements. As the auditor assumes more risk, moving from the 95 percent to the 85 percent nominal confidence level, the gap between the coverage desired and that actually provided by the modified DUS-cell bound (C2, C4) increases significantly, by approximately two and one-half times. This indicates that it would be very undesirable for the auditor to use the modified DUS-cell bound. Both these results indicate that the nominal confidence level chosen by the auditor can influence the performance of the DUS-cell bounds.

Research Question Seven

The seventh research question addresses the comparability of the results of this study to the results of the Helton study. Table 3.10 provides some descriptive characteristics of each population. As was noted in the previous chapter, the population (pop1) used for this study was an accounts payable population. The population (pop2) used in the Helton study was an accounts receivable population. A common assumption made by auditors about an accounts payable population is that errors tend to be understatements, whereas the the opposite is assumed for an accounts receivable population (4:390). Therefore, since

.....
 Table 3.10
 Descriptive Statistics of Book Values
 for the Accounting Population

	Pop 1	Pop 2
Number of Accounts	2986	7026
Mean Book Value	\$255.84	\$1,945.84
Standard Deviation	\$364.23	\$7,021.61
Minimum	\$1.85	\$.10
Maximum	\$8,110.66	\$98,162.70
Total	\$763,931.19	\$13,671,503.00
Variance	132,665.80	49,303,056.48
Skewness	8.524	7.944
Kurtosis	125.853	78.168

.....

the Helton study assumed that all errors would be overstatements, it would be expected that the Helton results would be better coverage and tighter bounds.

A comparison of the coverage of unmodified bounds indicates that the average mean coverage for each bound in the Helton study did outperform the coverage found in this study, as shown in Table 3.11. For the unmodified Stringer bound at both nominal confidence levels (S1, S3), the result was a robust bound, in both studies. However, in this study the unmodified DUS-cell bound at the 95 percent confidence level (C1) was found to be robust, while in the Helton study, it was not. Both studies found the unmodified DUS-

Table 3.11
Comparison of the Coverage of Unmodified Bounds

Bound	A Pop 1	B Pop 2	C Change (B-A)
S1 *	.9994	.9999	.0005
S3 *	.9883	.9915	.0032
C1 **	.9868	.9895	.0027
C3 **	.9240	[.9562]	.0322
* Stringer bound			
** DUS-cell bound			

cell bound at the 85 percent confidence level (C3) to lack robustness. Little significance is evident in any of these findings in that all differences are less than one percent.

A comparison of the coverage of the modified bounds indicates that the average mean coverage for each bound in the Helton study also outperformed the coverage found in this study, as shown in Table 3.12. For this study, one robust coverage resulted, the modified Stringer bound at the 85 percent confidence level (S4). The Helton study found that no modified bounds were robust. Yet the average mean overall coverage for the modified Stringer at the 85 percent confidence level was almost two percent better in the Helton study than that in this study. The Helton bounds performance overall was significantly better than this study, ranging from about one and a half percent to 11.6 percent.

A comparison of the relative tightness at the 95

Table 3.12
Comparison of the Coverage of Modified Bounds

Bound	A Pop 1	B Pop 2	C Change (B-A)
S2 *	.9415	.9922	.0507
S4 *	.9422	.9603	.0181
C2 **	.9415	[.9578]	[.0163]
C4 **	.7851	.9014	.1163
* Stringer bound			
** DUS-cell bound			

Table 3.13
Comparison of the Relative Tightness
for 95 Percent Nominal Confidence Level Bounds

Bound	A Pop 1	B Pop 2	C Change (B-A)
S1 *	1.0000	1.0000	-
C1 **	1.1429	[1.1442]	.0013
S2 *	1.1021	[1.1403]	.0382
C2 **	1.1333	1.1166	(.0167)
* Stringer bound			
** DUS-cell bound			

percent confidence level is shown in Table 3.13. The relative tightness of the modified DUS-cell bound at this nominal level of confidence (C2) shows this study's bound outperforming Helton's bounds examined in the Helton study. The unmodified DUS-cell (C1) and modified Stringer (S2) bounds at this nominal confidence level in the Helton study outperformed those corresponding bounds in this study. Similar results are found at the 85 percent confidence level, as shown by Table 3.14.

.....
 Table 3.14
 Comparison of the Relative Tightness
 for 85 Percent Nominal Confidence Level Bounds

DUS-cell Bound	A Pop 1	B Pop 2	C Change (B-A)
S3 *	1.0000	1.0000	.
C3 **	1.1033	[1.1068]	.0035
S4 *	1.0919	[1.1098]	.0179
C4 **	1.0958	1.0845	(.0113)
.....			
* Stringer bound			
** DUS-cell bound			
.....			

A comparison of the average percentage change for the DUS-cell bound is shown in Table 3.15. Again, the Helton study results outperformed the results of this study. All of the Helton study results were increases in overall average mean over the nominal confidence level. In this study, the modified DUS-cell bound at both nominal levels of confidence (C2, C4) were not robust overall. For the modified DUS-cell bound, the Helton study found the average percentage change of the 85 percent confidence level bound (C4) to be two times greater than the 95 percent confidence level bound (C2). This study found that difference to be about two and a half times less for the 85 percent confidence level bound (C4) to the 95 percent confidence level bound (C2). For the unmodified DUS-cell bounds (C1, C3), the Helton study found little difference between the two nominal confidence levels. This study found that the average percentage increase of the unmodified DUS-cell bound

.....
 Table 3.15
 Comparison of the Average Percentage Change of the
 Actual Confidence Level from the Nominal Confidence Level

DUS-cell Bound	A Pop 1	B Pop 2	C Change (B-A)
C1	.74	.79	.05
C2	(.17)	.16	.33
C3	.49	[.71]	.21
C4	(.43)	.34	.77

.....

at the 85 percent confidence level (C3) was about four times
 that of the unmodified DUS-cell bound at the 95 percent
 confidence level (C1).

IV. Summary and Conclusions

This chapter contains three major sections. The first of these sections is a summary of the findings of the research. The second section provides conclusions based upon these findings. The last section suggests areas of possible future research.

Summary

The accounts payable population used in this study contained more overstatements than understatements, an unexpected result. The small number of errors found in the population would not allow an accurate determination of the statistical distribution they best represented. Comparisons with three other studies indicate a need to examine different types of accounts using different statistical methods, since their statistical characteristics varied.

Through the simulation process, four bounds were analyzed, the unmodified Stringer, the modified Stringer, the unmodified DUS-cell, and the modified DUS-cell bounds. The unmodified Stringer bounds were found to be robust at both nominal levels of confidence, 85 and 95 percent. The unmodified DUS-cell bound at the 95 percent confidence level and the modified Stringer bound at the 85 percent level were also robust. When analyzing the data by EAI category for the unmodified DUS-cell bound at the 85 percent level of

confidence, both the medium and low EAI categories were robust. For the modified bounds, only the low EAI category yielded robust results for the modified Stringer at 95 percent confidence level and the modified DUS-cell bound at both confidence levels.

In a comparison with the unmodified Stringer bound, all other bounds at both confidence levels provided a tighter bound. The actual coverage provided by the unmodified DUS-cell bound exceeded the coverage required by the nominal confidence level at both levels. The modified DUS-cell bound did not provide enough coverage at either level.

In comparison with the results of the Helton study, differences were noted. The results were mixed; some bounds in this study outperformed those in the Helton study and vice versa. This study found more bounds to be robust, while the Helton study provided better average mean coverage at both levels of confidence. When comparing relative tightness, the results were split. The modified DUS-cell bound at both levels of confidence outperformed those in the Helton study, with the reverse being true for the modified Stringer and unmodified DUS-cell bounds at both levels of confidence. A comparison of the average percentage change of the actual confidence level from the nominal confidence level, Helton study bounds again outperformed those used in this study. This difference is particularly significant for the modified DUS-cell bound.

Conclusions

The findings of this study give auditors additional information for determining which bounds might be used in practice. The final determination will greatly depend upon how much risk auditors are willing to accept. The most conservative auditors would continue to use the unmodified Stringer bound. In both this study and the Helton study, this bound performed up to or exceeded expectations. If auditors are willing to accept more risk, then the choice may be the unmodified DUS-cell bound. In either instance, the bound outperforms the nominal confidence level. By choosing the DUS-cell bound, the auditor would also gain the benefit of bounds 10 to 14 percent relatively tighter on the average than the unmodified Stringer bound.

The modified bounds are not recommended for use. These bounds failed to perform consistently in either this study or the Helton study. Of particular significance was the lack of adequate coverage provided by the modified DUS-cell bound, .94 actual mean coverage for .95 nominal coverage and .79 actual mean coverage for .85 nominal coverage. Another contributor to this decision is that only at a low error amount intensity did this bound perform consistently near its nominal confidence level. This information coupled with the results from the Helton study strongly suggest using only the unmodified bounds, both the Stringer and the DUS-cell.

Based upon a comparison of the results of these two studies and little knowledge being available on the nature of errors in accounting populations, further research is suggested. This research should include new populations with different types of accounts. This would give auditors a better idea of how well the DUS-cell bound might perform.

Suggestions for Further Research

The following are suggestions for further research:

- A. How does the DUS-cell bound perform for inventory populations?
- B. Does the DUS-cell bound performance provide similar results for additional populations of the same type?
- C. How does the DUS-cell bound perform for populations with other error rates?

Appendix A: Computer Program 2

```
C*****
C      Sort Program in Ascending Numeric Order
C
C      Modification by: Steve Bringle
C
C      Key:
C          N = number of items
C          BLIST = real array to be sorted
C          SWITCH = 0 if sort complete; 1 if not complete
C          TEMP = temporary storage location
C*****
C
C      integer n, switch
C      real blist(2986), temp
C
C      open(7,file='in.dat')
C      open(8,file='i.dat')
C      rewind 7
C      rewind 8
C      switch = 1
C      n = 2986
C
C      read(8,*) (blist(j),j=1,n)
C      print *, (blist(j),j=1,n)
20  if(switch.eq.1) then
C          switch = 0
C          do 30 j = 1,n
C              if(blist(j).lt.blist(j-1)) then
C                  temp = blist(j)
C                  blist(j) = blist (j-1)
C                  blist(j-1) = temp
C                  switch = 1
C              endif
30      continue
C          go to 20
C      endif
C
C      write(7,50) (blist(j),j=1,n)
50  format(1x,f8.2)
C      end
```

Appendix B: Computer Program 3

```

C*****
C      AUTHOR:  MIKE HELTON
C
C      MODIFICATION: STEVE BRINGLE
C
C      This program is designed to randomly select line
C      items to put error in for my thesis.
C
C      Key
C      mid = midpoint of $ value
C      cnt = # of line items in bvary
C      sum = total $ amount
C      bvary(9000) = array with book values
C      ranary(7000) = array with book values to be
C      sampled
C      cntit = # of line items in low values
C      cntem = # of line items in high values
C      i,s,t,p are counters
C      ix,iy = values used in subroutine randu
C      yfl = # from subroutine between 0 and 1
C*****
C
C      Initialization of variables
C
C      real mid,bvary,yfl
C      real totary,errary,b,c
C      REAL BV,TOT
C      integer i,s,p,ix,iy,t,cnt,cntit,cntem,ranary,index
C      INTEGER J,M,G,P
C      dimension bvary(9000),errary(9000),totary(9000)
C      dimension ranary(7000),index(9000)
C
C      mid = 0.0
C      data i,s,p,t,cnt,cntit,cntem /7*0/
C      ix = 12345
C      yfl = 0.0
C      iy = 0
C
C      OPEN(1,FILE='IN1.DAT')
C      rewind 1
C      print *, 'the file is open and rewound'
C      open(2,file='sp04.dat')
C      rewind 2
C      i = 1
10 READ(1,*,END = 77) BV,TOT
   index(i) = i

```

```

i = i + 1
bvary(i) = bv
errary(i) = 0.0
totary(i) = tot
cnt = cnt + 1
go to 10
c
c
77 print *, 'the data has been read'
mid = totary(cnt)/2
print *, 'the mid value is', mid
i=1
12 if (mid .ge. totary(i)) then
    cntit = cntit + 1
    i = i + 1
    go to 12
else
    Print *, 'The mid value is', mid, 'and it occurs at
*line #', cntit
end if
c
    cntem = cnt - cntit
    print *, 'The cnt is', cnt, 'and cntit is', cntit, 'and
*cntem is', cntem
    p = 1
    j = 1
    b = 1.0
    ix = 12345
    call randu (ix, iy, yfl)
18 if (j .le. cntit .and. (b-1)/cntit .lt. .8) then
    if (index(j) .eq. 0) then
        j = j + 1
        if (j .gt. cntit) j=1
        go to 18
    end if
    ix = iy
    call randu (ix, iy, yfl)
    if (yfl .lt. .5) then
        ranary(p) = index(j)
        p = p + 1
        b = b + 1
        index(j) = 0
        j = j + 1
        if (j .gt. cntit) j=1
    else
        j = j + 1
        if (j .gt. cntit) j=1
    end if
    go to 18
end if
c
c

```

```

s = p - 1
c = b
t = 1 + cntit
19 if(t .ge. cntit .and. t .le. cnt .and. (c-p)/cntem
*.lt. .8) then
    if (index(t) .eq. 0) then
        t = t + 1
        if (t .gt. cnt) t = 1 + cntit
        go to 19
    end if
    ix = iy
    call randu(ix,iy,yfl)
    if (yfl .lt. .5) then
        s = s + 1
        ranary(s) = index(t)
        c = c + 1
        index(t) = 0
        t = t + 1
        if (t .gt. cnt) t = 1 + cntit
    else
        t = t + 1
        if (t .gt. cnt) t = 1 + cntit
    end if
    go to 19
end if
do 25 m=1,s
    print *,ranary(m)
25 continue
do 86 g=1,s
    write(2,15)ranary(g),cntit,cnt
15 format(1x,3(i6,2x))
86 continue
print *, 'S has a value of',s
close (1)
close (2)
end

```

c

```

subroutine randu(ix,iy,yfl)
INTEGER IX,IY
REAL YFL
iy = ix*65539
if (iy)5,6,6
5 iy = iy + 2147483647 + 1
6 yfl = iy
yfl = yfl*.4656613E-9
return
end

```

Appendix C: Computer Program 4

```

C      PROGRAM STUDY
C*****
C
C      PURPOSE:
C
C      REVISION: V2.0
C
C      DATE CREATED: 26 MAR 84
C      DATE REVISITED: 26 APR 84
C
C      AUTHOR: MAJ JEFF PHILLIPS
C
C      PROGRAM MODIFICATIONS: 1LT HAL STALCUP
C
C      REVISED: MIKE HELTON
C
C      MODIFICATION: STEVE BRINGLE
C*****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
*IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)
COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,
*SIMPER,TYSKP
common/study1/ zk,samper,sampbv,xmle,eff,effa,
*bp,bpa
common/study2/ pgw,pgwa,im,totant,itoot,igy,j,
*1,n
common/study3/ ix,iy,yfl,ns,fn
common/study4/ jzz,str,xmstr,stra,xmstra
common uel,uela,muel,muela
C*****
C      CALL TO SUBROUTINES BEGINS HERE:
C      INITA PUTS INITIAL VALUES INTO CONSTANT ARRAYS.
C      ERRATE PUTS ERRORS INTO STUDY POPULATIONS.
C*****
      open(1,file='belch.hat')
      open(4,file='belch.dat')
      open(7,file='in1.dat')
      open(9,file='belch.fat')

```

```

rewind 1
rewind 4
rewind 7
rewind 9
write(9,100)
100 format(/10x,'Begin simulation.....')
CALL INITA
CALL ERRATE
STOP
END

C
C
C *****
C SUBROUTINE SAMPL
C *****
C *****
C
C SAMPL IS USED TO ACCESS THE POPULATIONS AND PRODUCE A
C RANDOM SAMPLE FROM THE POPULATIONS. SAMPL ALSO
C CALCULATES THE BOUNDS USED BY THE NEXT SUBROUTINE.
C
C *****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
* IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)
COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,
* SIMPER,TYSKP
COMMON/STUDY1/ ZK, SAMPER, SAMPBV, XMLE, EFF, EFFA,
* BP, BPA
COMMON/STUDY2/ PGW, PGWA, IM, TOTANT, ITOOT, IGY, J,
* L, N
COMMON/STUDY3/ IX, IY, YFL, NS, FN
COMMON/STUDY4/ JZZ, STR, XMSTR, STRA, XMSTRA
COMMON UEL, UELA, MUEL, MUELA
C *****
IX=54321
CALL RANDU(IX,IY,YFL)
BP=3.00
BPA=1.90
SAMPBV=BIGBV
SAMPER=BIGER
XMLE=0.0

```

```

PGW=0.0
PGWA=0.0
EFF=0.0
EFA=0.0
IM=0
TOTANT=0.0
ITOOT=0
IGY=0
J=0
DO 4 JZZ=1,500
DO 57 ITO=1,350
A(ITO)=0.00
57 CONTINUE
C
CALL STEP1
C
IGY=0
DO 7 IK=1,N
SAMPER=SAMPER+EERARY(IK)
SAMPBV=SAMPBV+BBVARY(IK)
IF (TTNTRY(IK).NE.0.0) THEN
    ITOOT=ITOOT+1
    A(ITOOT)=TTNTRY(IK)
ELSE
    GO TO 7
END IF
7 CONTINUE
IF (ITOOT.LE.1) GO TO 33
C*****
C    SORT EACH SAMPLE IN DESCENDING ORDER BY TAINT
C*****
CALL SORTA(A,350,ITOOT)
C*****
C    CALCULATE THE STRINGER(STR) AND MODIFIED
C    STRINGER(MSTR) BOUNDS
C*****
33 continue
C
CALL BOUND1
C
C*****
C    CALCULATE THE CELL AND MODIFIED CELL BOUNDS
C*****
CALL BOUND2
C*****
C    OUTPUT RESULTS
C*****
ZNEG=TY-SAMPBV
WRITE(4,15) JJ,ISDYPP,JZZ,LL,MM,ITT,N,TE
WRITE(4,17) IM, TY, SAMPBV,SAMPER,SIMPER,ZNEG
17 FORMAT(1X,I5,2X,5(F12.2,2X))
15 FORMAT(1X,7(I4,2X),F12.2,2X)

```

```

      WRITE(4,16) STR,XMSTR,STRA,XMSTRA
      WRITE(4,16) UEL,MUEL,UELA,MUELA
16  FORMAT(1X,5(F14.2,2X))

```

C
C

```

      SAMPER=BIGER
      SAMPBV=BIGBV
      XMLE=0.0
      PGW=0.0
      PGWA=0.0
      IM=0
      ITOOT=0
      TOTANT=0.0
4  CONTINUE
      RETURN
      END

```

C
C
C

```

*****
      SUBROUTINE BOUND2
*****

```

C
C
C

C*****

```

      COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
      COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
      COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
      COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
      COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
      COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
      COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
      COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
      COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
      COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
* IENDD
      COMMON BBVARY(500),AAVARY(500),EERARY(500)
      COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,
* SIMPER,TYSKP
      COMMON/STUDY1/ ZK, SAMPER, SAMPBV, XMLE, EFF, EFFA,
* BP, BPA
      COMMON/STUDY2/ PGW, PGWA, IM, TOTANT, ITOOT, IGY, J,
* L, N
      COMMON/STUDY3/ IX, IY, YFL, NS, FN
      COMMON/STUDY4/ JZZ, STR, XMSTR, STRA, XMSTRA
      COMMON UEL, UELA, MUEL, MUELA

```

C*****

```

      real sumdt,las,lasa,dto,dtoa,ssv,ssva,suel,suela,fn
      real cumavg,dt,muela,muel
      integer it, iw, im
      sumdt=0.0
      las=0.0
      lasa=0.0

```

```

dto=1.0
dtoa=1.0
ssv=3.00
ssva=1.90
suel=3.00
suela=1.90
fn=0.0
muela=0.0
muel=0.0
sumb=0.0
sumba=0.0
im=0
do 12 it = 1,n
  if (a(it) .gt. 0.00) im = im + 1
12 CONTINUE

```

c
c

```

do 13 iw = 1,im
  sumb = sumb + b(iw)
  sumba = sumba + ba(iw)
  uelf = bp + iw + sumb
  uelfa = bpa + iw + sumba
  dt = a(iw)
  sumdt = sumdt + a(iw)
  cumavg = sumdt/iw
  if (ssv .gt. las) then
    uelp = ssv
  else
    uelp = las
  end if
  if(ssva .gt. lasa) then
    uelpa = ssva
  else
    uelpa = lasa
  end if
  las = uelp + a(iw)
  lasa = uelpa + a(iw)
  ssv = uelf*cumavg
  ssva = uelfa*cumavg
  if(ssv .gt. las) then
    suel = ssv
  else
    suel = las
  end if
  if (ssva .gt. lasa) then
    suela = ssva
  else
    suela = lasa
  end if
13 continue
fn = float(n)
uel = suel*tyskp/fn + simper

```

```

      uela = suela*tyskp/fn + simper
      muel = (ty - sampbv)*suel/fn + samper
      muela = (ty-sampbv)*suela/fn + samper
      return
      end

```

C
C
C

```

*****

```

```

      SUBROUTINE BOUND1

```

C
C
C

```

*****

```

C*****

```

      COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
      COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
      COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
      COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
      COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
      COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
      COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
      COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
      COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
      COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,

```

```

      *IENDD

```

```

      COMMON BBVARY(500),AAVARY(500),EERARY(500)

```

```

      COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,

```

```

      *SIMPER,TYSKP

```

```

      COMMON/STUDY1/ ZK, SAMPER, SAMPBV, XMLE, EFF, EFFA,

```

```

      *BP, BPA

```

```

      COMMON/STUDY2/ PGW, PGWA, IM, TOTANT, ITOOT, IGY, J,

```

```

      *L, N

```

```

      COMMON/STUDY3/ IX, IY, YFL, NS, FN

```

```

      COMMON/STUDY4/ JZZ, STR, XMSTR, STRA, XMSTRA

```

```

      common uel,uela,muel,muela

```

C*****

```

      DO 10 IZ=1,N

```

```

      XMLE=XMLE+A(IZ)

```

```

      PGW=PGW+(B(IZ)*A(IZ))

```

```

      PGWA=PGWA+(BA(IZ)*A(IZ))

```

```

10 CONTINUE

```

```

      FN = FLOAT(N)

```

```

      STR=(BP+XMLE+PGW)*TYSKP/FN+SIMPER

```

```

      STRA=(BPA+XMLE+PGWA)*TYSKP/FN+SIMPER

```

```

      XMSTR=(TY-SAMPBV)*(BP+XMLE+PGW)/FN+SAMPER

```

```

      XMSTRA=(TY-SAMPBV)*(BPA+XMLE+PGWA)/FN+SAMPER

```

```

      RETURN

```

```

      END

```

C
C

```

*****

```

```

      SUBROUTINE STEP1

```

C
C

```

*****

```

```

C*****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
* IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)
COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,
* SIMPER,TYSKP
COMMON/STUDY1/ ZK, SAMPER, SAMPBV, XMLE, EFF, EFFA,
* BP, BPA
COMMON/STUDY2/ PGW, PGWA, IM, TOTANT, ITOOT, IGY, J,
* L, N
COMMON/STUDY3/ IX, IY, YFL, NS, FN
COMMON/STUDY4/ JZZ, STR, XMSTR, STRA, XMSTRA
common uel,uela,muel,muela
C*****
14 IX=IY
CALL RANDU(IX,IY,YFL)
IF (YFL.EQ.0.0) GO TO 14
Y=YFL*SKIP
DO 6 IMY=1,N
    TIJ=Y+(SKIP*(FLOAT(IMY)-1.0))
    CALL SEARCH(TIJ,LINEI)
    IGY=IGY+1
    BBVARY(IGY)=BVARY(LINEI)
    AAVARY(IGY)=AVARY(LINEI)
    EERARY(IGY)=ERRARY(LINEI)
    TTNTRY(IGY)=TNTARY(LINEI)
    TTOTRY(IGY)=TOTARY(LINEI)
    IINDEX(IGY)=INDEX(LINEI)
6 CONTINUE
C    WRITE(1,100) JZZ,IGY
C    DO 60 IGH=1,IGY
C    WRITE(1,200) BBVARY(IGH),AAVARY(IGH),ERRARY(IGH),
C    *TTNTRY(IGH),TTOTRY(IGH),IINDEX(IGH)
C 60 CONTINUE
C 100 FORMAT(10X,'INTEGRATION=',I5,'    SAMPLE SIZE=',I5)
C 200 FORMAT(5X,6F12.2)
    RETURN
    END

C
C *****
C    SUBROUTINE SEARCH(TIJ,LINEI)
C *****
C

```

```

C*****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
* IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)
COMMON TTNTARY(500),TTOTRY(500),IINDEX(500),SKIP,
* SIMPER,TYSKP
COMMON/STUDY1/ ZK, SAMPER, SAMPBV, XMLE, EFF, EFFA,
* BP, BPA
COMMON/STUDY2/ PGW, PGWA, IM, TOTANT, ITOOT, IGY, J,
* L, N
COMMON/STUDY3/ IX, IY, YFL, NS, FN
COMMON/STUDY4/ JZZ, STR, XMSTR, STRA, XMSTRA
common uel,uela,muel,muela
C*****
INTEGER MAXPT, MINPT, MIDPT
C
MINPT = 1
MAXPT = IENDD
10 CONTINUE
MIDPT = (MAXPT-MINPT)/2 + MINPT
IF ( (MAXPT-MINPT) .EQ.1) THEN
    LINEI = MAXPT
    RETURN
ELSE
    IF (TIJ.LE.TOTARY(MIDPT)) THEN
        MAXPT = MIDPT
    ELSE
        MINPT = MIDPT
    ENDIF
ENDIF
GO TO 10
END
C
C *****
C SUBROUTINE ERRATE
C *****
C
C *****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)

```

```

COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
* IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)
COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,
* SIMPER,TYSKP
common/study1/ zk,samper,sampbv,xmle,eff,effa,
* bp,bpa
common/study2/ pgw,pgwa,im,totant,itoot,igy,j,
* l,n
common/study3/ ix,iy,yfl,ns,fn
common/study4/ jzz,str,xmstr,stra,xmstra
common uel,uela,muel,muela
C*****
C JJ IS POPULATIONS 1 THRU 4 (ORDERED POPULATIONS)
C LL IS ERROR RATES BY LINE ITEM (.50,.30,.15,.01)
C MM IS ERROR RATE DISTRIBUTION (UNIFORM,DECREASING,
C INCREASING)
C ITT IS TAINTINGS (.40,.20 AND .20,.10)
C KK IS STRATA (STRATA 1=LOW BV'S;STRATA 2=HIGH BV'S)
C*****
integer hasit
integer r
dimension hasit(9000)
J=0
I=0
KNT=0
KUZ=0
KAT=0
MID=0
IENDD=0
EEXP=0.0
ISDYPP=0
IX=12345
CALL RANDU(IX,IY,YFL)
DO 532 JJ=1,1
IF (JJ.EQ.1) then
write(9,201)
201 format(/lx,'Opened in1, processing.....')
endif
c IF (JJ.EQ.2) then
c OPEN(7,FILE='pop2.dat')
c write(9,202)
c 202 format(/lx,'Opened pop2, processing.....')
c endif
c IF (JJ.EQ.3) then
c OPEN(7,FILE='pop3.dat')
c write(9,203)

```

```

c 203   format(/lx,'Opened pop3, processing.....')
c       endif
c       IF (JJ.EQ.4) then
c         OPEN(7,FILE='pop4.dat')
c         write(9,204)
c 204   format(/lx,'Opened pop4, processing.....')
c       endif
c
c
c
c       54 read(7,*,end=55) bv,tot
c
c       I=I+1
c       BVARV(I)=BV
c       ERRARY(I)=0.0
c       TOTARY(I)=TOT
c       AVARY(I)=0.0
c       TNTARY(I)=0.0
c       INDEX(I)=I
c       IRND(I)=0
c       GO TO 54
c 55 CONTINUE
c       CLOSE(7)
c       write(9,200) i
c 200   format(/lx,'Processed ',i5,' records.')
c       IF (JJ.EQ.1) then
c         OPEN(8,FILE='sp04.dat')
c         write(9,301)
c 301   format(/lx,'Opened sp04, processing.....')
c       endif
c       IF (JJ.EQ.2) then
c         CPEN(8,FILE='sp05.dat')
c         write(9,302)
c 302   format(/lx,'Opened sp05, processing.....')
c       endif
c       IF (JJ.EQ.3) then
c         OPEN(8,FILE='sp06.dat')
c         write(9,303)
c 303   format(/lx,'Opened sp06, processing.....')
c       endif
c       IF (JJ.EQ.4) then
c         OPEN(8,FILE='sp02.dat')
c         write(9,304)
c 304   format(/lx,'Opened sp02, processing.....')
c       endif
c       REWIND 8
c       I=0
c 999 read(8,*,end=64) iiw,mid,iendd
c       i = i + 1
c       IRND(IIW)=IIW
c       GO TO 999
c 64 CONTINUE
c       write(9,300) i

```

```

300 format(/lx,'Processed ',i5,' records.')
i = 0
CLOSE(8)
DO 533 LL=1,4
C   IF (JJ.EQ.1) THEN
C   IF (LL.EQ.1) THEN
C       PR(1)=.50684
C       PR(2)=.25342
C       PR(3)=.50
C       PR(4)=.486855
C       PR(5)=.973710
END IF
C   IF (LL.EQ.2) THEN
C       PR(1)=.304105
C       PR(2)=.152053
C       PR(3)=.30
C       PR(4)=.292113
C       PR(5)=.584226
END IF
C   IF (LL.EQ.3) THEN
C       PR(1)=.152053
C       PR(2)=.076026
C       PR(3)=.15
C       PR(4)=.146056
C       PR(5)=.292113
END IF
C   IF (LL.EQ.4) THEN
C       PR(1)=.010137
C       PR(2)=.005068
C       PR(3)=.01
C       PR(4)=.009737
C       PR(5)=.019474
END IF
C   END IF
C   IF (JJ.EQ.2) THEN
C   IF (LL.EQ.1) THEN
C       PR(1)=.520888
C       PR(2)=.260444
C       PR(3)=.50
C       PR(4)=.462877
C       PR(5)=.925754
END IF
C   IF (LL.EQ.2) THEN
C       PR(1)=.312532
C       PR(2)=.156266
C       PR(3)=.30
C       PR(4)=.277726
C       PR(5)=.555452
END IF
C   IF (LL.EQ.3) THEN
C       PR(1)=.156266
C       PR(2)=.078133

```

```

C          PR(3)=.15
C          PR(4)=.138863
C          PR(5)=.277726
C      END IF
C      IF (LL.EQ.4) THEN
C          PR(1)=.010418
C          PR(2)=.005209
C          PR(3)=.01
C          PR(4)=.009258
C          PR(5)=.018515
C      END IF
C  END IF
C  IF (JJ.EQ.3) THEN
C      IF (LL.EQ.1) THEN
C          PR(1)=.512059
C          PR(2)=.256029
C          PR(3)=.50
C          PR(4)=.477509
C          PR(5)=.955019
C      END IF
C      IF (LL.EQ.2) THEN
C          PR(1)=.307235
C          PR(2)=.153618
C          PR(3)=.30
C          PR(4)=.286506
C          PR(5)=.573011
C      END IF
C      IF (LL.EQ.3) THEN
C          PR(1)=.153618
C          PR(2)=.076809
C          PR(3)=.15
C          PR(4)=.143253
C          PR(5)=.286506
C      END IF
C      IF (LL.EQ.4) THEN
C          PR(1)=.010241
C          PR(2)=.005121
C          PR(3)=.01
C          PR(4)=.009550
C          PR(5)=.019100
C      END IF
C  END IF
C  IF (JJ.EQ.4) THEN
C      IF (LL.EQ.1) THEN
C          PR(1)=.508408
C          PR(2)=.254204
C          PR(3)=.50
C          PR(4)=.483992
C          PR(5)=.967984
C      END IF
C      IF (LL.EQ.2) THEN
C          PR(1)=.305045

```

```

C          PR(2)=.152522
C          PR(3)=.30
C          PR(4)=.290395
C          PR(5)=.580790
C      END IF
C      IF (LL.EQ.3) THEN
C          PR(1)=.152522
C          PR(2)=.076261
C          PR(3)=.15
C          PR(4)=.145197
C          PR(5)=.290395
C      END IF
C      IF (LL.EQ.4) THEN
C          PR(1)=.010168
C          PR(2)=.005084
C          PR(3)=.01
C          PR(4)=.009679
C          PR(5)=.019359
C      END IF
C      END IF
C      DO 534 MM=1,3
C      IF (MM.EQ.1) THEN
C          p=pr(3)
C          pp=pr(3)
C      END IF
C      IF (MM.EQ.2) THEN
C          p=pr(3)
C          pp=pr(3)
C      END IF
C      IF (MM.EQ.3) THEN
C          p=pr(3)
C          pp=pr(3)
C      END IF
C      DO 405 ITT=1,2
C      IF (ITT.EQ.1) THEN
C          XMOO(1)=.8131
C          XMOO(2)=.2083
C      ELSE
C          XMOO(1)=.2083
C          XMOO(2)=.10
C      END IF
C      KNT=0
C      KUZ=0
C      KAT=0
C      do 8 r = 1,9000
C      hasit(r)=0
8  continue
C      DO 502 KK=1,IENDD
C      ix = iy
C      call randu(ix,iy,yfl)
C      if (yfl .lt. .5) go to 502
C      if (hasit(kk) .gt. 0) go to 502

```

```

IZZ=INT(P*0.1)
IF (KK.LE.MID) THEN
    IJKLM=INT(FLOAT(MID)*P)
ELSE
    IJKLM=INT(FLOAT(IENDD-MID)*PP)
END IF
IF (KK.LE.MID) THEN
    IF ((KNT.LE.IJKLM).AND.(IRND(KK).NE.0)) THEN
        IF (KUZ.LE.IZZ) THEN
            AVARY(KK)=BVARY(KK)+(BVARY(KK)*1.0)
            KUZ=KUZ+1
            KNT=KNT+1
        ELSE
1000      IX=IY
            CALL RANDU(IX,IY,YFL)
            IF (YFL.EQ.0.0) GO TO 1000
            EEXP=-XMOO(1)*ALOG(YFL)
            IF (EEXP.GE.1.0) GO TO 1000
            AVARY(KK)=BVARY(KK)+(BVARY(KK)*EEXP)
            KNT=KNT+1
        END IF
    ELSE
        AVARY(KK)=BVARY(KK)
    END IF
END IF
IF (KK.GT.MID) THEN
2000      IF ((KAT.LE.IJKLM).AND.(IRND(KK).NE.0)) THEN
            IX=IY
            CALL RANDU(IX,IY,YFL)
            IF (YFL.EQ.0.0) GO TO 2000
            EEXP=-XMOO(2)*ALOG(YFL)
            IF (EEXP.GE.1.0) GO TO 2000
            AVARY(KK)=BVARY(KK)+(BVARY(KK)*EEXP)
            KAT=KAT+1
        ELSE
            AVARY(KK)=BVARY(KK)
        END IF
    END IF
    ERRARY(KK)=AVARY(KK)-BVARY(KK)
    TNTARY(KK)=ERRARY(KK)/BVARY(KK)
    if(kk .ge. mid .and. knt .lt. int(float(mid)*p)) kk=1
    if(kk .eq. iendd .and. kat .lt. ijkml) kk = mid + 1
    hasit(kk) = kk
502  CONTINUE
    ISDYPP=ISDYPP+1
    CALL INFO
    CALL STRPOF
    write(9,600)
600  format(1x,'Processing sample, please standby.....')
    CALL SAMPL
405  CONTINUE
534  CONTINUE

```

```

533 CONTINUE
    close(4)
532 CONTINUE
    RETURN
    END

```

C
C
C
C

```

*****
SUBROUTINE RANDU(IX,IY,YFL)
*****
    integer ix,iy
    real yfl
    IY=IX*65539
    IF (IY)5,6,6
5 IY=IY+2147483647+1
6 YFL=IY
  YFL=YFL*.4656613E-9
  RETURN
  END

```

C
C
C
C
C
C

```

*****
SUBROUTINE SORTA(A,ND,NS)
*****
    integer ns,nd
    REAL A(ND), TEMP
    INTEGER I, LASTS, LASTI, SSTART
    LOGICAL INSORT

```

C

```

    SSTART = NS - 1
    LASTS = 1
    LASTI = LASTS
    INSORT = .FALSE.
10 CONTINUE
    IF (.NOT.INSORT) THEN
        INSORT = .TRUE.
        DO 20 I = SSTART, LASTI, -1
            IF (A(I).LT.A(I+1)) THEN
                TEMP = A(I)
                A(I) = A(I+1)
                A(I+1) = TEMP
                INSORT = .FALSE.
                LASTS = I
            ENDIF
        CONTINUE
        LASTI = LASTS+1
        GO TO 10
    ENDIF
    RETURN

```

```

END
C
C
C
C *****
C SUBROUTINE INFO
C *****
C *****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
*IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)
COMMON TTNTRY(500),TTOTRY(500),IINDEX(500),SKIP,
*SIMPER,TYSKP
common/study1/ zk,samper,sampbv,xmle,eff,effa,
*bp,bpa
common/study2/ pgw,pgwa,im,totant,itoot,igy,j,
*1,n
common/study3/ ix,iy,yfl,ns,fn
common/study4/ jzz,str,xmstr,stra,xmstra
common uel,uela,muel,muela
C*****
C M IS # OF ERRORS
C NX IS # OF LINE ITEMS
C TY IS STUDY POPULATION TOTAL BOOK VALUE
C TYE IS STUDY POPULATION TOTAL BOOK VALUE IN ERROR
C TE IS TOTAL ERROR VALUE IN STUDY POPULATION
C*****
NX=0
TE=0.0
TY=0.0
M=0
TYE=0.0
TELI=0.0
TEDV=0.0
TBR=0.0
TETYY=0.0
DO 1 I=1,IENDD
NX=NX+1
TE=TE+ERRARY(I)
TY=TY+BVARAY(I)
IF (ERRARY(I).NE.0.0) THEN
M=M+1

```


common/study4/ jzz,str,xmstr,stra,xmstra
common uel,uela,muel,muela

C*****

B(1)=.75
B(2)=.55
B(3)=.46
B(4)=.40
B(5)=.36
B(6)=.33
B(7)=.30
B(8)=.29
B(9)=.27
B(10)=.26
B(11)=.24
B(12)=.24
B(13)=.22
B(14)=.22
B(15)=.21
B(16)=.21
B(17)=.19
B(18)=.20
B(19)=.18
B(20)=.19
B(21)=.18
B(22)=.17
B(23)=.17
B(24)=.17
B(25)=.16
B(26)=.16
B(27)=.16
B(28)=.15
B(29)=.16
B(30)=.15
B(31)=.15
B(32)=.15
B(33)=.15
B(34)=.15
B(35)=.15
B(36)=.15
B(37)=.15
B(38)=.15
B(39)=.15
B(40)=.13
B(41)=.13
B(42)=.13
B(43)=.13
B(44)=.13
B(45)=.12
B(46)=.12
B(47)=.12
B(48)=.12
B(49)=.12

B(50)=.12
B(51)=.12
B(52)=.12
B(53)=.12
B(54)=.12
B(55)=.12
B(56)=.12
B(57)=.12
B(58)=.12
B(59)=.12
B(60)=.11
B(61)=.11
B(62)=.11
B(63)=.11
B(64)=.11
B(65)=.10
B(66)=.10
B(67)=.10
B(68)=.10
B(69)=.10
B(70)=.10
B(71)=.10
B(72)=.10
B(73)=.10
B(74)=.10
B(75)=.10
B(76)=.10
B(77)=.10
B(78)=.10
B(79)=.10
B(80)=.09
B(81)=.09
B(82)=.09
B(83)=.09
B(84)=.09
B(85)=.09
B(86)=.09
B(87)=.09
B(88)=.09
B(89)=.09
B(90)=.09
B(91)=.09
B(92)=.09
B(93)=.09
B(94)=.09
B(95)=.08
B(96)=.08
B(97)=.08
B(98)=.08
B(99)=.08
B(100)=.08
B(101)=.08

B(102)=.08
B(103)=.08
B(104)=.08
B(105)=.08
B(106)=.07
B(107)=.07
B(108)=.07
B(109)=.07
B(110)=.07
B(111)=.07
B(112)=.07
B(113)=.07
B(114)=.07
B(115)=.07
BA(1)=.48
BA(2)=.35
BA(3)=.29
BA(4)=.25
BA(5)=.23
BA(6)=.21
BA(7)=.19
BA(8)=.18
BA(9)=.17
BA(10)=.17
BA(11)=.15
BA(12)=.15
BA(13)=.14
BA(14)=.14
BA(15)=.13
BA(16)=.13
BA(17)=.13
BA(18)=.12
BA(19)=.12
BA(20)=.11
BA(21)=.11
BA(22)=.11
BA(23)=.11
BA(24)=.11
BA(25)=.10
BA(26)=.10
BA(27)=.10
BA(28)=.10
BA(29)=.10
BA(30)=.09
BA(31)=.09
BA(32)=.09
BA(33)=.09
BA(34)=.09
BA(35)=.09
BA(36)=.09
BA(37)=.09
BA(38)=.09

BA(39)=.09
BA(40)=.09
BA(41)=.08
BA(42)=.08
BA(43)=.08
BA(44)=.08
BA(45)=.08
BA(46)=.08
BA(47)=.08
BA(48)=.08
BA(49)=.08
BA(50)=.07
BA(51)=.07
BA(52)=.07
BA(53)=.07
BA(54)=.07
BA(55)=.07
BA(56)=.07
BA(57)=.07
BA(58)=.07
BA(59)=.07
BA(60)=.07
BA(61)=.07
BA(62)=.07
BA(63)=.07
BA(64)=.07
BA(65)=.07
BA(66)=.07
BA(67)=.07
BA(68)=.07
BA(69)=.07
BA(70)=.06
BA(71)=.06
BA(72)=.06
BA(73)=.06
BA(74)=.06
BA(75)=.06
BA(76)=.06
BA(77)=.06
BA(78)=.06
BA(79)=.06
BA(80)=.06
BA(81)=.06
BA(82)=.06
BA(83)=.06
BA(84)=.06
BA(85)=.06
BA(86)=.06
BA(87)=.06
BA(88)=.06
BA(89)=.06
BA(90)=.05

```

BA(91)=.05
BA(92)=.05
BA(93)=.05
BA(94)=.05
BA(95)=.05
BA(96)=.05
BA(97)=.05
BA(98)=.05
BA(99)=.05
BA(100)=.05
BA(101)=.05
BA(102)=.05
BA(103)=.05
BA(104)=.05
BA(105)=.05
BA(106)=.04
BA(107)=.04
BA(108)=.04
BA(109)=.04
BA(110)=.04
BA(111)=.04
BA(112)=.04
BA(113)=.04
BA(114)=.04
BA(115)=.04
DO 55 JXYZ=116,250
    B(JXYZ)=0.0
    BA(JXYZ)=0.0
55  CONTINUE
    end
C *****
C   SUBROUTINE STRPOF
C *****
C
C *****
C   THIS SUBROUTINE IDENTIFIES LINE ITEMS OF THE
C   POPULATION WITH BOOK VALUES GREATER THAN THE
C   SAMPLING SKIP INTERVALS
C *****
COMMON A(350),B(350),TBV(20),M,NX,TY,JJ,LL,MM,ITT
COMMON XLTLA(11),XLTLB(11),PI(11),TYE,TE
COMMON XMU(11),CS(11),XMCS(11),F(200,11),FA(200,11)
COMMON YTR2CS(11),YCS(11),YTR4CS(11),YMCS(11)
COMMON BVARY(9000),ERRARY(9000),IYI(9000),IYM(9000)
COMMON XTR1CS(11),XTR2CS(11),XTR3CS(11),XTR4CS(11)
COMMON TELI,TEDV,TBR,BA(350),IRND(9000),BIGBV,BIGER
COMMON IYJ(9000),IYK(9000),IYL(9000),INDEX(9000)
COMMON TOTARY(9000),XMDPT(20),MIDL(20),IENDPT(20)
COMMON PR(5),XMOO(2),AVARY(9000),TNTARY(9000),ISDYPP,
* IENDD
COMMON BBVARY(500),AAVARY(500),EERARY(500)

```

```

COMMON TTTRY(500),TTOTRY(500),IINDEX(500),SKIP,
*SIMPER,TYSKP
COMMON/STUDY1/ ZK, SAMPER, SAMPBV, XMLE, EFF, EFFA,
*BP, BPA
COMMON/STUDY2/ PGW, PGWA, IM, TOTANT, ITOOT, IGY, J,
*L, N
COMMON/STUDY3/ IX, IY, YFL, NS, FN
COMMON/STUDY4/ JZZ, STR, XMSTR, STRA, XMSTRA
common uel,uela,muel,muela
C*****
N=200
JJUMP=0
IJUMP=0
IBANG=0
BIGBV=0.0
IJACK=IENDD
BIGER=0.0
SIMPER=0.0
TYSKP=TY
33 SKIP=TYSKP/N
IBANG=0
DO 10 I=1,IJACK
IF (BVARY(I).GT.SKIP) THEN
J=I
IBANG=1
GO TO 34
END IF
10 CONTINUE
IF (IBANG.EQ.0) GO TO 35
34 DO 20 K=J,IJACK
IJUMP=IJUMP+1
JJUMP=JJUMP+1
BIGBV=BIGBV+BVARV(K)
BIGER=BIGER+ERRARY(K)
SIMPER=SIMPER+ERRARY(K)
20 CONTINUE
TYSKP=TY-BIGBV
N=N-JJUMP
IJACK=IENDD-IJUMP
JJUMP=0
IF (IBANG.GT.0) GO TO 33
35 RETURN
END

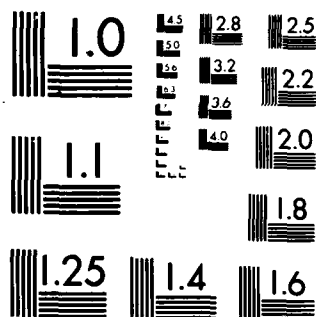
```

Appendix D: Computer Program 5

```
c   Author: Jeff Phillips
c
c   Revised: Mike Helton
c
c   *****
c   program exam1
c   *****
c
c   symbols for the various bounds are as follows:
c       95% confidence
c   s1-stringer
c   s2-modified stringer
c
c   c1-cell
c   c2-modified cell
c
c       85% confidence
c   s3-stringer
c   s4-modified stringer
c
c   c3-cell
c   c4-cell bound
c
c *****
c   common ipop,jj,irep,ier,ied,itnt,n,te
c   common im,ty,sampbv,samper,simper,zneg
c   common s1(500),s2(500),s3(500),s4(500)
c   common c1(500),c2(500),c3(500),c4(500)
c   common ii,k,nxx
c   common cvs1,cvs2,cvs3,cvs4
c   common cvc1,cvc2,cvc3,cvc4
c   common rls1,rls2,rls3,rls4
c   common rlc1,rlc2,rlc3,rlc4
c   common r2s1,r2s2,r2s3,r2s4
c   common r2c1,r2c2,r2c3,r2c4
c   common r3s1,r3s2,r3s3,r3s4
c   common r3c1,r3c2,r3c3,r3c4
c   common xminsl,xmins2,xmins3,xmins4
c   common xmincl,xminc2,xminc3,xminc4
c   common qls1,qls2,qls3,qls4
c   common qlc1,qlc2,qlc3,qlc4
c   common xmeds1,xmeds2,xmeds3,xmeds4
c   common xmedcl,xmedc2,xmedc3,xmedc4
c   common xmns1,xmns2,xmns3,xmns4
c   common xmnc1,xmnc2,xmnc3,xmnc4
c   common q3s1,q3s2,q3s3,q3s4
c   common q3c1,q3c2,q3c3,q3c4
c   common xmaxs1,xmaxs2,xmaxs3,xmaxs4
c   common xmaxcl,xmaxc2,xmaxc3,xmaxc4
```

AD-A174 172 A STUDY OF UPPER ERROR LIMITS IN ACCOUNTING POPULATIONS 2/2
(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH
SCHOOL OF SYSTEMS AND LOGISTICS G S BRINGLE SEP 86
UNCLASSIFIED AFIT/BSM/LSV/86S-4 F/G 12/1 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```

common sds1,sds2,sds3,sds4
common sdc1,sdc2,sdc3,sdc4
c*****
c
c   main program begins here.
c
c   open(4,file='belch.dat')
c   open(7,file='analysis')
c   rewind 4
c   rewind 7
c   do 5 ii=4,4
c
c   ii-four(4) main populations
c
c   do 10 jj=1,24
c
c   twenty-four(24) study pops per main population
c
c   call input
c   call cvr2r3
c   call sort
c   call script
10 continue
5 continue
close(7)
stop
end
c
c
c   call to subroutine begins here.
c   *****
c   subroutine input
c   *****
c
c   this subroutine puts data into arrays and computes
c   accumulators for "coverage, rel tightness two &
c   three"
c*****
common ipop,jj,irep,ier,ied,itnt,n,te
common im,ty,sampbv,samper,simper,zneg
common sl(500),s2(500),s3(500),s4(500)
common cl(500),c2(500),c3(500),c4(500)
common ii,k,nxx
common cvs1,cvs2,cvs3,cvs4
common cvc1,cvc2,cvc3,cvc4
common rls1,rls2,rls3,rls4
common rlc1,rlc2,rlc3,rlc4
common r2s1,r2s2,r2s3,r2s4
common r2c1,r2c2,r2c3,r2c4
common r3s1,r3s2,r3s3,r3s4
common r3c1,r3c2,r3c3,r3c4
common xmins1,xmins2,xmins3,xmins4

```

```

common xminc1,xminc2,xminc3,xminc4
common qls1,qls2,qls3,qls4
common qlc1,qlc2,qlc3,qlc4
common xmeds1,xmeds2,xmeds3,xmeds4
common xmedc1,xmedc2,xmedc3,xmedc4
common xmns1,xmns2,xmns3,xmns4
common xmnc1,xmnc2,xmnc3,xmnc4
common q3s1,q3s2,q3s3,q3s4
common q3c1,q3c2,q3c3,q3c4
common xmaxs1,xmaxs2,xmaxs3,xmaxs4
common xmaxc1,xmaxc2,xmaxc3,xmaxc4
common sds1,sds2,sds3,sds4
common sdc1,cdc2,cdc3,cdc4
C*****
  call vzero
  do 20 k=1,500
    nxx=200
    read(ii,15) ipop,isdyp,irep,ier,ied,itnt,n,te
    read(ii,17) im,ty,sampbv,samper,simper,zneg
    read(ii,16) s1(k),s2(k),s3(k),s4(k)
    read(ii,16) c1(k),c2(k),c3(k),c4(k)
15  format(1x,7(i4,2x),f12.2,2x)
16  format(1x,5(f14.2,2x))
17  format(1x,i5,2x,5(f12.2,2x))
C
C
C    coverage accumulators start here.
C
    if (s1(k).ge.te) cvs1=cvs1+1.0
    if (s2(k).ge.te) cvs2=cvs2+1.0
    if (s3(k).ge.te) cvs3=cvs3+1.0
    if (s4(k).ge.te) cvs4=cvs4+1.0
C
    if (c1(k).ge.te) cvc1=cvc1+1.0
    if (c2(k).ge.te) cvc2=cvc2+1.0
    if (c3(k).ge.te) cvc3=cvc3+1.0
    if (c4(k).ge.te) cvc4=cvc4+1.0
C
C
C    rel tightness two(2) accumulators start here.
C
    r2s1=r2s1+(s1(k)/s1(k))
    r2s2=r2s2+(s2(k)/s2(k))
    r2s3=r2s3+(s3(k)/s3(k))
    r2s4=r2s4+(s4(k)/s4(k))
C
    r2c1=r2c1+(s1(k)/c1(k))
    r2c2=r2c2+(s2(k)/c2(k))
    r2c3=r2c3+(s3(k)/c3(k))
    r2c4=r2c4+(s4(k)/c4(k))
C
  20 continue
  return
end
C*****

```

```

C
C
C *****
C      subroutine cvr2r3
C *****
C
C      subroutine calculates coverage, rel tightness
C      two(2)
C *****
      common ipop,jj,irep,ier,ied,itnt,n,te
      common im,ty,sampbv,samper,simper,zneg
      common s1(500),s2(500),s3(500),s4(500)
      common c1(500),c2(500),c3(500),c4(500)
      common ii,k,nxx
      common cvs1,cvs2,cvs3,cvs4
      common cvc1,cvc2,cvc3,cvc4
      common rls1,rls2,rls3,rls4
      common rlc1,rlc2,rlc3,rlc4
      common r2s1,r2s2,r2s3,r2s4
      common r2c1,r2c2,r2c3,r2c4
      common r3s1,r3s2,r3s3,r3s4
      common r3c1,r3c2,r3c3,r3c4
      common xmins1,xmins2,xmins3,xmins4
      common xminc1,xminc2,xminc3,xminc4
      common qls1,qls2,qls3,qls4
      common qlc1,qlc2,qlc3,qlc4
      common xmeds1,xmeds2,xmeds3,xmeds4
      common xmedc1,xmedc2,xmedc3,xmedc4
      common xmns1,xmns2,xmns3,xmns4
      common xmnc1,xmnc2,xmnc3,xmnc4
      common q3s1,q3s2,q3s3,q3s4
      common q3c1,q3c2,q3c3,q3c4
      common xmaxs1,xmaxs2,xmaxs3,xmaxs4
      common xmaxc1,xmaxc2,xmaxc3,xmaxc4
      common sds1,sds2,sds3,sds4
      common sdc1,sdc2,sdc3,sdc4
C *****
      x=500.0
C
C      coverage calculations begin here.
C
      cvs1=cvs1/x
      cvs2=cvs2/x
      cvs3=cvs3/x
      cvs4=cvs4/x
C
      cvc1=cvc1/x
      cvc2=cvc2/x
      cvc3=cvc3/x
      cvc4=cvc4/x
C
C      rel tightness two(2) calculations begin here.

```

```

C      r2s1=r2s1/x
      r2s2=r2s2/x
      r2s3=r2s3/x
      r2s4=r2s4/x

C      r2c1=r2c1/x
      r2c2=r2c2/x
      r2c3=r2c3/x
      r2c4=r2c4/x

C      return
      end

C
C
C      *****
      subroutine sort
      *****

C      subroutine sorts all bounds array from lowest to
C      highest value and calls subroutine statrl.
C      *****
      common ipop,jj,irep,ier,ied,itnt,n,te
      common im,ty,sampbv,samper,simper,zneg
      common s1(500),s2(500),s3(500),s4(500)
      common c1(500),c2(500),c3(500),c4(500)
      common ii,k,nxx
      common cvs1,cvs2,cvs3,cvs4
      common cvc1,cvc2,cvc3,cvc4
      common rls1,rls2,rls3,rls4
      common rlc1,rlc2,rlc3,rlc4
      common r2s1,r2s2,r2s3,r2s4
      common r2c1,r2c2,r2c3,r2c4
      common r3s1,r3s2,r3s3,r3s4
      common r3c1,r3c2,r3c3,r3c4
      common xminsl,xmins2,xmins3,xmins4
      common xmincl,xmnc2,xmnc3,xmnc4
      common qls1,qls2,qls3,qls4
      common qlc1,qlc2,qlc3,qlc4
      common xmeds1,xmeds2,xmeds3,xmeds4
      common xmedcl,xmedc2,xmedc3,xmedc4
      common xmns1,xmns2,xmns3,xmns4
      common xmnc1,xmnc2,xmnc3,xmnc4
      common q3s1,q3s2,q3s3,q3s4
      common q3c1,q3c2,q3c3,q3c4
      common xmaxs1,xmaxs2,xmaxs3,xmaxs4
      common xmaxcl,xmaxc2,xmaxc3,xmaxc4
      common sds1,sds2,sds3,sds4
      common sdc1,cdc2,cdc3,cdc4
C      *****
      nn=500
      mnl=nn-1

```

```

do 1 j=1,nml
nmj=nn-j
do 2 i=1,nmj
if (s1(i).gt.s1(i+1)) then
tmps1=s1(i)
s1(i)=s1(i+1)
s1(i+1)=tmps1
end if
c
if (s2(i).gt.s2(i+1)) then
tmps2=s2(i)
s2(i)=s2(i+1)
s2(i+1)=tmps2
end if
c
if (s3(i).gt.s3(i+1)) then
tmps3=s3(i)
s3(i)=s3(i+1)
s3(i+1)=tmps3
end if
c
if (s4(i).gt.s4(i+1)) then
tmps4=s4(i)
s4(i)=s4(i+1)
s4(i+1)=tmps4
end if
c
if (c1(i).gt.c1(i+1)) then
tmpc1=c1(i)
c1(i)=c1(i+1)
c1(i+1)=tmpc1
end if
c
if (c2(i).gt.c2(i+1)) then
tmpc2=c2(i)
c2(i)=c2(i+1)
c2(i+1)=tmpc2
end if
c
if (c3(i).gt.c3(i+1)) then
tmpc3=c3(i)
c3(i)=c3(i+1)
c3(i+1)=tmpc3
end if
c
if (c4(i).gt.c4(i+1)) then
tmpc4=c4(i)
c4(i)=c4(i+1)
c4(i+1)=tmpc4
end if
c
2 continue

```

```

1 continue
  call statr1
  return
end
c *****
  subroutine statr1
c *****
c
c   subroutine computes statistics for each bound and
c   calculates rel tightness one(1)
c*****
  common ipop,jj,irep,ier,ied,itnt,n,te
  common im,ty,sampbv,samper,simper,zneg
  common s1(500),s2(500),s3(500),s4(500)
  common c1(500),c2(500),c3(500),c4(500)
  common ii,k,nxx
  common cvs1,cvs2,cvs3,cvs4
  common cvc1,cvc2,cvc3,cvc4
  common rls1,rls2,rls3,rls4
  common rlc1,rlc2,rlc3,rlc4
  common r2s1,r2s2,r2s3,r2s4
  common r2c1,r2c2,r2c3,r2c4
  common r3s1,r3s2,r3s3,r3s4
  common r3c1,r3c2,r3c3,r3c4
  common xminsl,xmins2,xmins3,xmins4
  common xmincl,xminc2,xminc3,xminc4
  common qls1,qls2,qls3,qls4
  common qlc1,qlc2,qlc3,qlc4
  common xmeds1,xmeds2,xmeds3,xmeds4
  common xmedcl,xmedc2,xmedc3,xmedc4
  common xmns1,xmns2,xmns3,xmns4
  common xmnc1,xmnc2,xmnc3,xmnc4
  common q3s1,q3s2,q3s3,q3s4
  common q3c1,q3c2,q3c3,q3c4
  common xmaxs1,xmaxs2,xmaxs3,xmaxs4
  common xmaxcl,xmaxc2,xmaxc3,xmaxc4
  common sds1,sds2,sds3,sds4
  common sdc1,cdc2,cdc3,cdc4
c*****
  nn=500
c
  xmeds1=(s1(nn/2)+s1((nn/2)+1))/2.0
  xmeds2=(s2(nn/2)+s2((nn/2)+1))/2.0
  xmeds3=(s3(nn/2)+s3((nn/2)+1))/2.0
  xmeds4=(s4(nn/2)+s4((nn/2)+1))/2.0
c
  xmedcl=(c1(nn/2)+c1((nn/2)+1))/2.0
  xmedc2=(c2(nn/2)+c2((nn/2)+1))/2.0
  xmedc3=(c3(nn/2)+c3((nn/2)+1))/2.0
  xmedc4=(c4(nn/2)+c4((nn/2)+1))/2.0
c
  xminsl=s1(1)

```

```

xmins2=s2(1)
xmins3=s3(1)
xmins4=s4(1)
c
xminc1=c1(1)
xminc2=c2(1)
xminc3=c3(1)
xminc4=c4(1)
c
xmaxs1=s1(nn)
xmaxs2=s2(nn)
xmaxs3=s3(nn)
xmaxs4=s4(nn)
c
maxc1=c1(nn)
maxc2=c2(nn)
maxc3=c3(nn)
maxc4=c4(nn)
c
qls1=s1(nn/4)
qls2=s2(nn/4)
qls3=s3(nn/4)
qls4=s4(nn/4)
c
qlc1=c1(nn/4)
qlc2=c2(nn/4)
qlc3=c3(nn/4)
qlc4=c4(nn/4)
c
q3s1=s1(nn*3/4)
q3s2=s2(nn*3/4)
q3s3=s3(nn*3/4)
q3s4=s4(nn*3/4)
c
q3c1=c1(nn*3/4)
q3c2=c2(nn*3/4)
q3c3=c3(nn*3/4)
q3c4=c4(nn*3/4)
c
do 3 i=1,500
c
xmns1=xmns1+s1(i)
xmns2=xmns2+s2(i)
xmns3=xmns3+s3(i)
xmns4=xmns4+s4(i)
c
xmnc1=xmnc1+c1(i)
xmnc2=xmnc2+c2(i)
xmnc3=xmnc3+c3(i)
xmnc4=xmnc4+c4(i)
c
3 continue

```

```

      xi=500.0
c
      xmns1=xmns1/xi
      xmns2=xmns2/xi
      xmns3=xmns3/xi
      xmns4=xmns4/xi
c
      xmnc1=xmnc1/xi
      xmnc2=xmnc2/xi
      xmnc3=xmnc3/xi
      xmnc4=xmnc4/xi
c
      do 4 i=1,500
c
      sigs1=sigs1+((s1(i)-xmns1))**2
      sigs2=sigs2+((s2(i)-xmns2))**2
      sigs3=sigs3+((s3(i)-xmns3))**2
      sigs4=sigs4+((s4(i)-xmns4))**2
c
      sigc1=sigc1+((c1(i)-xmnc1))**2
      sigc2=sigc2+((c2(i)-xmnc2))**2
      sigc3=sigc3+((c3(i)-xmnc3))**2
      sigc4=sigc4+((c4(i)-xmnc4))**2
c
4 continue
c
      sds1=(sigs1/(xi-1))**0.5
      sds2=(sigs2/(xi-1))**0.5
      sds3=(sigs3/(xi-1))**0.5
      sds4=(sigs4/(xi-1))**0.5
c
      sdc1=(sigc1/(xi-1))**0.5
      sdc2=(sigc2/(xi-1))**0.5
      sdc3=(sigc3/(xi-1))**0.5
      sdc4=(sigc4/(xi-1))**0.5
c
      rls1=xmeds1/xmeds1
      rls2=xmeds2/xmeds2
      rls3=xmeds3/xmeds3
      rls4=xmeds4/xmeds4
c
      rlc1=xmeds1/xmedc1
      rlc2=xmeds2/xmedc2
      rlc3=xmeds3/xmedc3
      rlc4=xmeds4/xmedc4
c
      return
      end
c
      *****
c      subroutine script
c      *****

```

```

c
c      subroutine writes output to file called
c      analysis
c*****
      common ipop,jj,irep,ier,ied,itnt,n,te
      common im,ty,sampbv,samper,simper,zneg
      common sl(500),s2(500),s3(500),s4(500)
      common cl(500),c2(500),c3(500),c4(500)
      common ii,k,nxx
      common cvsl,cvs2,cvs3,cvs4
      common cvcl,cvc2,cvc3,cvc4
      common rls1,rls2,rls3,rls4
      common rlc1,rlc2,rlc3,rlc4
      common r2s1,r2s2,r2s3,r2s4
      common r2c1,r2c2,r2c3,r2c4
      common r3s1,r3s2,r3s3,r3s4
      common r3c1,r3c2,r3c3,r3c4
      common xmins1,xmins2,xmins3,xmins4
      common xmincl,xminc2,xminc3,xminc4
      common qls1,qls2,qls3,qls4
      common qlc1,qlc2,qlc3,qlc4
      common xmeds1,xmeds2,xmeds3,xmeds4
      common xmedc1,xmedc2,xmedc3,xmedc4
      common xmns1,xmns2,xmns3,xmns4
      common xmnc1,xmnc2,xmnc3,xmnc4
      common q3s1,q3s2,q3s3,q3s4
      common q3c1,q3c2,q3c3,q3c4
      common xmaxs1,xmaxs2,xmaxs3,xmaxs4
      common xmaxc1,xmaxc2,xmaxc3,xmaxc4
      common sds1,sds2,sds3,sds4
      common sdc1,cdc2,cdc3,cdc4
c*****
      write(7,*)
      write(7,*)
      write(7,301) ipop,jj,ier,ied,itnt,nxx,te
      write(7,*)
      write(7,302) cvsl,cvc2,cvs3,cvs4
      write(7,302) cvcl,cvc2,cvc3,cvc4
      write(7,*)
      write(7,302) rls1,rls2,rls3,rls4
      write(7,302) rlc1,rlc2,rlc3,rlc4
      write(7,*)
      write(7,302) r2s1,r2s2,r2s3,r2s4
      write(7,302) r2c1,r2c2,r2c3,r2c4
      write(7,*)
      write(7,303) xmins1,xmins2,xmins3,xmins4
      write(7,303) xmincl,xminc2,xminc3,xminc4
      write(7,*)
      write(7,303) qls1,qls2,qls3,qls4
      write(7,303) qlc1,qlc2,qlc3,qlc4
      write(7,*)
      write(7,303) xmns1,xmns2,xmns3,xmns4

```

```

write(7,303) xmnc1,xmnc2,xmnc3,xmnc4
write(7,*)
write(7,303) xmeds1,xmeds2,xmeds3,xmeds4
write(7,303) xmedc1,xmedc2,xmedc3,xmedc4
write(7,*)
write(7,303) q3s1,q3s2,q3s3,q3s4
write(7,303) q3c1,q3c2,q3c3,q3c4
write(7,*)
write(7,303) xmaxs1,xmaxs2,xmaxs3,xmaxs4
write(7,303) xmaxc1,xmaxc2,xmaxc3,xmaxc4
write(7,*)
write(7,303) sds1,sds2,sds3,sds4
write(7,303) sdc1,cdc2,cdc3,cdc4
301 format(1x,6(i4,2x),f12.2,2x)
302 format(1x,5(f12.4,2x))
303 format(1x,5(f12.2,2x))
return
end

c
c
c *****
c      subroutine vzero
c *****
c
c      subroutine initializes and zeros out all variables
c*****
common ipop,jj,irep,ier,ied,itnt,n,te
common im,ty,sampbv,samper,simper,zneg
common s1(500),s2(500),s3(500),s4(500)
common c1(500),c2(500),c3(500),c4(500)
common ii,k,nxx
common cvs1,cvs2,cvs3,cvs4
common cvc1,cvc2,cvc3,cvc4
common rls1,rls2,rls3,rls4
common rlc1,rlc2,rlc3,rlc4
common r2s1,r2s2,r2s3,r2s4
common r2c1,r2c2,r2c3,r2c4
common r3s1,r3s2,r3s3,r3s4
common r3c1,r3c2,r3c3,r3c4
common xmin1,xmin2,xmin3,xmin4
common xminc1,xminc2,xminc3,xminc4
common qls1,qls2,qls3,qls4
common qlc1,qlc2,qlc3,qlc4
common xmeds1,xmeds2,xmeds3,xmeds4
common xmedc1,xmedc2,xmedc3,xmedc4
common xmns1,xmns2,xmns3,xmns4
common xmnc1,xmnc2,xmnc3,xmnc4
common q3s1,q3s2,q3s3,q3s4
common q3c1,q3c2,q3c3,q3c4
common xmaxs1,xmaxs2,xmaxs3,xmaxs4
common xmaxc1,xmaxc2,xmaxc3,xmaxc4
common sds1,sds2,sds3,sds4

```

```

      common sdc1,sdc2,sdc3,sdc4
c*****
c
      ipop=0
      irep=0
      ier=0
      ied=0
      itnt=0
      n=0
      te=0.0
c
      do 1 i=1,500
      s1(i)=0.0
      s2(i)=0.0
      s3(i)=0.0
      s4(i)=0.0
c
      c1(i)=0.0
      c2(i)=0.0
      c3(i)=0.0
      c4(i)=0.0
c
      1 continue
c
      cvs1=0.0
      cvs2=0.0
      cvs3=0.0
      cvs4=0.0
c
      cvc1=0.0
      cvc2=0.0
      cvc3=0.0
      cvc4=0.0
c
      rls1=0.0
      rls2=0.0
      rls3=0.0
      rls4=0.0
c
      rlc1=0.0
      rlc2=0.0
      rlc3=0.0
      rlc4=0.0
c
      r2s1=0.0
      r2s2=0.0
      r2s3=0.0
      r2s4=0.0
c
      r2c1=0.0
      r2c2=0.0
      r2c3=0.0

```

```

r2c4=0.0
c
xmins1=0.0
xmins2=0.0
xmins3=0.0
xmins4=0.0
c
xminc1=0.0
xminc2=0.0
xminc3=0.0
xminc4=0.0
c
qls1=0.0
qls2=0.0
qls3=0.0
qls4=0.0
c
qlc1=0.0
qlc2=0.0
qlc3=0.0
qlc4=0.0
c
xmeds1=0.0
xmeds2=0.0
xmeds3=0.0
xmeds4=0.0
c
xmedc1=0.0
xmedc2=0.0
xmedc3=0.0
xmedc4=0.0
c
xmns1=0.0
xmns2=0.0
xmns3=0.0
xmns4=0.0
c
xmnc1=0.0
xmnc2=0.0
xmnc3=0.0
xmnc4=0.0
c
q3s1=0.0
q3s2=0.0
q3s3=0.0
q3s4=0.0
c
q3c1=0.0
q3c2=0.0
q3c3=0.0
q3c4=0.0
c

```

```

xmaxs1=0.0
xmaxs2=0.0
xmaxs3=0.0
xmaxs4=0.0
c
xmaxc1=0.0
xmaxc2=0.0
xmaxc3=0.0
xmaxc4=0.0
c
sigsl=0.0
sigsl2=0.0
sigsl3=0.0
sigsl4=0.0
c
sigcl=0.0
sigcl2=0.0
sigcl3=0.0
sigcl4=0.0
c
sds1=0.0
sds2=0.0
sds3=0.0
sds4=0.0
c
sdcl=0.0
sdcl2=0.0
sdcl3=0.0
sdcl4=0.0
c
return
end

```

Bibliography

1. Ageloff, Roy and Richard Mojena. Applied Fortran 77. Belmont, CA: Wadsworth Publishing Company, 1981.
2. Andrews, Wesley T., Jr. and Alan G. Mayper. "Dollar-Unit Sampling," The Internal Auditor, 40:31-34 (April 1983).
3. Cox, D. R. and E. J. Snell. "On Sampling and the Estimation of Rare Errors," Biometrika, 66(1):125-132 (September 1979).
4. Ham, Jane, Donna Losell, and Wally Smieliauskas. "An Empirical Study Of Error Characteristics in Accounting Populations," The Accounting Review 60(3):387-406 (July 1985).
5. Helton, 1Lt Michael W. A Validation of an Accounting Upper Error Limit Bound. MS Thesis, AFIT/GSM/LSY/85S-17, School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1985.
6. Johnson, Johnny R., Robert A. Leitch, and John Neter. "Characteristics of Errors in Accounts Receivable and Inventory Audits," The Accounting Review, 56(2):270-293 (April 1981).
7. Leslie, Donald A., Albert D. Teitlebaum, and Rodney J. Anderson. Dollar Unit Sampling: A Practical Guide for Auditors. Toronto: Copp Clark Pittman Publishers Inc., 1979.
8. Neter, John, Johnny R. Johnson, and Robert A. Leitch. "Characteristics of Dollar-Unit Taints and Error Rates in Accounts Receivable and Inventory," The Accounting Review, 60(3):488-499 (July 1985).
9. Neter, John, Robert A. Leitch and Stephen E. Fienberg. "Dollar Unit Sampling: Multinomial Bounds for Total Overstatement and Understatement Errors," The Accounting Review, 53(1):77-93 (January 1978).
10. Neter, John and James K. Loebbecke. "Behavior of Major Statistical Estimators in Sampling Accounting Populations," New York: American Institute of Certified Public Accountants, Inc., 1975.

11. Neter, John and James K. Loebbecke. "On the Behavior of Statistical Estimators When Sampling Accounting Populations," Journal of American Statistical Association, 72(359):501-507 (September 1977).
12. Norusis, Marija J. SPSSx Introductory Statistics Guide. New York: McGraw-Hill Book Company, 1983.
13. SPSSx Basics. New York: McGraw-Hill Book Company, 1984.

VITA

Captain G. Steven Bringle was born 30 March 1950 at Fort George G. Meade, Maryland. He served in the enlisted ranks of the United States Air Force from July, 1972 until his honorable discharge in June, 1976 at the rank of Sergeant. After his service, he attended the University of Central Florida from which he received the degree of Bachelor of Science in Business Administration in August, 1979. He also received his commission through the ROTC program at that time. Upon graduation, he was assigned to the Space Launch and Control Systems SPO at Headquarters Space Division, Los Angeles AFS, California. In July, 1983, he was assigned to the F-15 SPO at Headquarters Aeronautical Systems Division, Wright-Patterson AFB, Ohio and served there until entering the School of Systems and Logistics, Air Force Institute of Technology, in May, 1985.

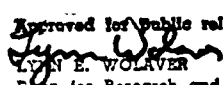
Permanent address: 828 Chipley Court
Winter Park, Florida 32792

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD A174 72

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GSM/LSY/86S-4			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION School of Systems and Logistics		6b. OFFICE SYMBOL (If applicable) AFIT/LSY	7a. NAME OF MONITORING ORGANIZATION		
6c. ADDRESS (City, State and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433-6583			7b. ADDRESS (City, State and ZIP Code)		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER		
8c. ADDRESS (City, State and ZIP Code)			10. SOURCE OF FUNDING NOS.		
			PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.
					WORK UNIT NO.
11. TITLE (Include Security Classification) See Box 19					
12. PERSONAL AUTHOR(S) G. Steven Bringle, B.S., Capt, USAF					
13a. TYPE OF REPORT MS Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Yr., Mo., Day) 1986 September	
				15. PAGE COUNT 113	
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB. GR.	Accounting, Auditing, Sampling, Model Theory		
05	01				
14	01				
19. ABSTRACT (Continue on reverse if necessary and identify by block number)					
Title: A STUDY OF UPPER ERROR LIMITS IN ACCOUNTING POPULATIONS					
Thesis Chairman: Jeffrey J. Phillips, Lt Col, USAF Assistant Professor of Accounting					
<div style="text-align: right;"> <p>Approved for public release; DOW AFB 100-W.  Lynn E. Wolaver Dean for Research and Professional Development Air Force Institute of Technology (AFIT) Wright-Patterson AFB OH 45433</p> </div>					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED		
22a. NAME OF RESPONSIBLE INDIVIDUAL Jeffrey J. Phillips, Lt Col, USAF			22b. TELEPHONE NUMBER (Include Area Code) 513-255-4845		22c. OFFICE SYMBOL AFIT/LSY

The purpose of this research was to examine a new accounts payable accounting population, comparing it to other populations which have been studied, examine the validity of an upper error limit bound, and compare those results with the results of a previous study. The bound examined was the Leslie, Teitlebaum, and Anderson DUS-cell bound. This method was supposed to reduce the bound conservatism and produce actual confidence levels closer to the nominal confidence levels.

The analysis of the DUS-cell bound was accomplished by examining the robustness, the relative tightness, and the effects of error amount intensity on the coverage provided and the relative tightness of the bound. The analysis of the other areas was by comparison.

The results of the research indicate that statistical characteristics varied for different accounting populations. The analysis of the validity of the DUS-cell bound method indicate that it is robust at the 95 percent confidence level, but is not at the 85 percent confidence level. In both cases, the DUS-cell bound is tighter than the Stringer bound. The results also indicate that error amount intensity significantly affects the coverage and the relative tightness provided by the DUS-cell bound. Comparing the results to a previous study of the validity of the bound provides mixed results. For this reason, further research needs to be accomplished in this area.

END

12-86

DTIC